

PATENT ABSTRACTS OF JAPAN

(11)Publication number : 2001-188686

(43)Date of publication of application : 10.07.2001

(51)Int.Cl. G06F 11/00

G06F 12/00

G06F 12/14

G06F 15/02

G06K 17/00

G06K 19/07

(21)Application number : 2000-294387 (71)Applicant : SONY CORP

(22)Date of filing : 27.09.2000 (72)Inventor : IIDA KENICHI

NOGUCHI MARI

OBAYASHI HIDEJI

(30)Priority

Priority number : 11301335

Priority date : 22.10.1999

Priority country : JP

(54) DATA REWRITING DEVICE, CONTROL METHOD, AND RECORDING MEDIUM

(57)Abstract:

PROBLEM TO BE SOLVED: To provide a data rewriting device without necessitating an external device, suitable for a portable terminal and capable of realizing an efficient processing, and also a control method and a recording medium.

SOLUTION: This data rewriting device rewrites data stored in a flash ROM 2 as a nonvolatile memory the rewriting of which is restricted which is provided to a portable information terminal main body 20 by using a program to execute update to be supplied from a plate like memory 7 as an attachable/detachable recording medium and the data to be used for update.

LEGAL STATUS [Date of request for examination]

[Date of sending the examiner's decision of rejection]

[Kind of final disposal of application other than the examiner's decision of rejection or application converted registration]

[Date of final disposal for application]

[Patent number]

[Date of registration]

[Number of appeal against examiner's decision of rejection]

[Date of requesting appeal against examiner's decision of rejection]

[Date of extinction of right]

* NOTICES *

JPO and NCIP are not responsible for any damages caused by the use of this translation.

1.This document has been translated by computer. So the translation may not reflect the original precisely.

2.*** shows the word which can not be translated.

3.In the drawings, any words are not translated.

CLAIMS

[Claim(s)]

[Claim 1] The updating data for updating the data memorized inside data rewriting equipment, In data rewriting equipment with the removable record medium with which

the discernment data which identify that program data, and the above-mentioned updating data and program data for transmitting the above-mentioned updating data are recorded on the above-mentioned data rewriting equipment were recorded the above -- with the 1st storage means which the program data for transmitting the above-mentioned updating data are read into the above-mentioned data rewriting equipment recorded on the above-mentioned record medium, and is remembered to be an insertion means to insert a removable record medium in the body of equipment A detection means to detect the above-mentioned discernment data from the above-mentioned record medium, and 2nd storage means by which are in the interior of the above-mentioned rewriting equipment, and the above-mentioned data by which updating is carried out are memorized, When the above-mentioned discernment data are detected by the above-mentioned detection means from the above-mentioned record medium After reading the program data for transmitting the above-mentioned updating data to the above-mentioned data rewriting equipment based on the discernment data by which detection was carried out [above-mentioned] from the above-mentioned record medium and making the storage means of the above 1st memorize, Data rewriting equipment characterized by having the control means which updates the data which read the above-mentioned updating data recorded on the above-mentioned record medium based on the program data memorized by the storage means of the above 1st, and are memorized by the 2nd storage means inside the above-mentioned data rewriting equipment.

[Claim 2] It is data rewriting equipment characterized by being program data for the above-mentioned updating data operating the above-mentioned data rewriting equipment in data rewriting equipment according to claim 1.

[Claim 3] It is data rewriting equipment characterized by to have further 3rd storage means to by which the load program data which the above-mentioned data rewriting equipment transmits [data] the program data for transmitting the above-mentioned updating data currently recorded on the above-mentioned record medium to the storage means of the above 1st in data rewriting equipment according to claim 1, and make it memorize were memorized.

[Claim 4] It is data rewriting equipment characterized by to cancel regulation of rewriting with the program data which were further equipped with a regulation means regulate rewriting of the data with which the above-mentioned data rewriting equipment is memorized inside the above-mentioned data rewriting equipment in data rewriting equipment according to claim 1, and were memorized by the above-mentioned storage means.

[Claim 5] Data rewriting equipment characterized by attaining discharge of regulation of the above-mentioned rewriting after reading the program data for transmitting the above-mentioned updating data to the above-mentioned data rewriting equipment in data rewriting equipment according to claim 2.

[Claim 6] The data memorized inside the above-mentioned data rewriting equipment in data rewriting equipment according to claim 1 are data rewriting equipment characterized by what is memorized by the nonvolatile memory means.

[Claim 7] The updating data for updating the data which the above-mentioned data rewriting equipment is further equipped with the 4th storage means, and are memorized inside the above-mentioned data rewriting equipment in data rewriting equipment according to claim 1 are data rewriting equipment which carries out [being used in order to update the data memorized by the 2nd storage means inside the above-mentioned data rewriting equipment after memorized by the storage means of the above 4th, and] as the description.

[Claim 8] It is data rewriting equipment characterized by for the above-mentioned control means to search the 1st location data in which the location where the program data for transmitting the above-mentioned updating data to the above-mentioned data rewriting equipment are recorded on the above-mentioned record medium is shown, and to read the above-mentioned program data in data rewriting equipment according to claim 1 based on the 1st location data by which retrieval was carried out [above-mentioned].

[Claim 9] It is data rewriting equipment which carries out [that the above-mentioned control means reads the above-mentioned updating data with which the updating data for updating the data memorized inside the above-mentioned data rewriting equipment searched the 2nd location data in which the location currently recorded on the above-mentioned record medium is shown, and were recorded on the above-mentioned record medium based on the 2nd location data by which retrieval was carried out / above-mentioned / in data rewriting equipment according to claim 1, and] as the description.

[Claim 10] The updating data for updating the data memorized inside data rewriting equipment, The program data for transmitting the above-mentioned updating data to the above-mentioned data rewriting equipment, From the above-mentioned body of data rewriting equipment with which the discernment data to identify were recorded, and a removable record medium, that the above-mentioned updating data and program data are recorded In the control approach of transmitting and storing the above-mentioned updating data in the above-mentioned data rewriting equipment The detection step which detects the above-mentioned discernment data from the above-mentioned record medium, and when discernment data are detected from the above-mentioned record medium The read-out step which reads the above-mentioned program data from the above-mentioned record medium based on the discernment data by which detection was carried out [above-mentioned], The updating data for updating the data memorized inside the above-mentioned data rewriting equipment from the above-mentioned record medium based on the program data by which reading appearance was carried out [above-mentioned] are read. The

control approach characterized by having the updating step which updates the data memorized inside the above-mentioned data rewriting equipment by the updating data by which reading appearance was carried out [above-mentioned].

[Claim 11] The control approach characterized by making the temporary storage means with which the above-mentioned data rewriting equipment is equipped memorize the above-mentioned program data in the control approach according to claim 10.

[Claim 12] It is the control approach characterized by having further the temporary storage step which makes the updating data recorded on a temporary storage means by which the above-mentioned data rewriting equipment is equipped with the above-mentioned control approach in the control approach according to claim 10 by the above-mentioned record medium memorize temporarily.

[Claim 13] The detection step which detects the above-mentioned discernment data of the above-mentioned control approach in the control approach according to claim 10 is the control approach characterized by detecting the above-mentioned discernment data by whether there is any predetermined data pattern for every block of the predetermined data size which constitutes the above-mentioned record medium.

[Claim 14] The updating step which updates the data of the above-mentioned control approach in the control approach according to claim 10 is the control approach characterized by updating data after eliminating the data of the location which was memorized by the above-mentioned data rewriting equipment, and which is updated.

[Claim 15] In the control approach according to claim 10 the above-mentioned control approach The step which reads the updating data for updating the data memorized inside data rewriting equipment from the above-mentioned record medium is preceded. The above-mentioned updating data are further equipped with the retrieval step which searches the 1st location data in which the location currently recorded on the above-mentioned record medium is shown. The control approach characterized by reading the updating data for updating the data memorized inside data rewriting equipment based on the location data of the above 1st from the above-mentioned record medium.

[Claim 16] It is the control approach which carries out [that the above-mentioned control approach is further equipped with the retrieval step which searches the 2nd location data in which the location where the program data for transmitting the above-mentioned updating data to the above-mentioned data rewriting equipment are recorded on the above-mentioned record medium is shown in the control approach according to claim 10, and read-out of the above-mentioned program data is performed based on the location data of the above 2nd, and] as the description.

[Claim 17] The record medium had the updating data storage area where the updating data for updating the data memorized inside the data rewriting equipment with which it

is equipped are recorded, the program data storage area where the program data for transmitting the above-mentioned updating data are recorded on the above-mentioned data rewriting equipment, and the discernment data storage area where the discernment data identify that the above-mentioned updating data and program data are recorded are recorded.

[Claim 18] The record medium characterized by having further the 1st location record section where the 1st location data in which the location where the above-mentioned updating data are recorded on the above-mentioned record medium is shown in a record medium according to claim 17 is recorded.

[Claim 19] The record medium characterized by having further the 2nd location record section where the 2nd location data in which the location where the above-mentioned program data are recorded on the above-mentioned record medium is shown in a record medium according to claim 17 is recorded.

[Claim 20] The above-mentioned record medium is a record medium characterized by having the management data record section where the management data which manages the data recorded on the above-mentioned record medium is recorded apart from the 1st location record section where the location data of the above 1st are recorded in a record medium according to claim 18.

DETAILED DESCRIPTION

[Detailed Description of the Invention]

[0001]

[Field of the Invention] The invention in this application relates to the data rewriting equipment, the control approach, and record medium which perform renewal of the data recorded on the nonvolatile memory with which equipment was equipped, and to which it rewrote and the limit was carried out based on the program and updating data for updating which are supplied by the above-mentioned equipment from a removable record medium.

[0002]

[Description of the Prior Art] now , in the microcomputer built into the interior of Personal Digital Assistant equipment , there be some which have the function which can rewrite from the outside the data which be needed at the time of actuation of the program for control the equipment which added the function of equipment even after mount in a substrate , or be memorized by the flash memory for the purpose of improvement in the engine performance , or a program by use as a flash memory the ROM data storage area which store a program . Usually, in order to realize rewriting from serial interface or the equipment outside which it has as a part of function to

these microcomputers, a serial communication interface is prepared in equipment, and it enables it to rewrite ROM data from external write-in equipment using serial communication.

[0003] Generally, since it is the very dangerous activity which opts for actuation of the equipment, rewriting actuation of the ROM data for these functional addition or the improvement in functional is constituted so that it may not go into a write-in condition easily by mistake. Conventionally therefore, at the time of ROM data rewriting in a service center, a repair section, etc. In order to start rewriting actuation, since ~~** is opened outside equipment~~ and it changes into the condition accessible in each part article inside equipment The hard connection for transmitting a required control signal for example, by soldering a jumper or being made to carry out according to the physical activity of a switch of the above-mentioned connection, a switch of a switch, etc. He was trying not to start rewriting actuation of ROM data accidentally by the operation mistake according the program which rewrites the ROM data memorized by equipment to a user etc.

[0004] Moreover, with the navigation equipment for automobiles, navigation equipment is equipped with CD-ROM for rewriting of ROM data etc., and it is made to perform correspondence to renewal of the data used for an addition and navigation of a new function by rewriting ROM data with the programme loader formed in the navigation equipment whose power source is stable.

[0005] Moreover, since it corresponds, for example to new Internet Protocol or a network application program with the router equipment of the Internet, Or in order to newly add the program for protecting LAN (Local Area Network) from cracking etc. The rewriting file of ROM data is supplied to router equipment from a server by the program which downloads the file for update which it has in the interior of router equipment. The programme loader which similarly was formed in the router equipment whose power source is stable is made to perform rewriting actuation of ROM data.

[0006] furthermore, the small storage carrying solid-state storage elements, such as recent years, for example, a flash memory etc., -- forming -- the drive equipment of dedication -- or drive equipment is built in an audio/video equipment, information machines and equipment, etc., and the thing which enables it to memorize computer data, image data, audio data, etc. is developed.

[0007]

[Problem(s) to be Solved by the Invention] However, the rewriting activity of the ROM data mentioned above having un-arrange [that it be necessary to disassemble and perform equipment by the serviceman, a service provider etc. or, and it be necessary that need to prepare the fixture of dedication and it be necessary to connect with the predetermined external terminal of equipment even if it be the case where the user himself rewrite a program], in order to switch the electric circuit of the equipment for make it for ROM data to write and not replace accidentally from an operation mistake

etc.

[0008] Moreover, to the personal digital assistant equipment whose power source is not stable, actuation of a microcomputer became unstable by rapid fluctuation, a rapid fall, etc. of supply voltage, and since there was a possibility that the program which rewrites ROM data by overrun may operate in being the worst, there was un-arranging [that the programme loader for rewriting of ROM data could not be formed in equipment].

[0009] Moreover, in the storage system using such a solid-state storage element, performing processing to the memorized file efficiently is called for. For example, it is required that that the data migration, the duplicate, rewriting, etc. which is needed for file processing within a storage are the minimum, the processing time, and power consumption should be made into the minimum.

[0010] On the other hand, since FAT (File Allocation Table) is used as a file system, the development environment of the program of the ROM data which many mentioned above is convenient for a ROM copy of data, if ROM data are described as file systems, such as FAT, also in case ROM data are written in a microcomputer through a record medium. However, since it is necessary to interpret comparatively large-scale file systems, such as FAT, by the equipment side in this case, processing of record data becomes very complicated.

[0011] Specifically, it is needed in the memory access driver section in equipment using the logic physical address translation table for changing the physical address used when actually accessing the logical address and the flash memory which are treated in the logic space which conversion of the address has for example, on a flash memory between the physical address for using it for physical access to a flash memory, and the logical address for processing data in logic space using FAT. Moreover, the elimination unit of a flash memory writes in and a means to process this efficiently on the assumption that the physical specification of being larger than a unit is needed. Furthermore, in FAT, since a data access is performed on the assumption that the record medium which can overwrite data, in case a flash memory with the need of writing in after eliminating is used, it is necessary to the write request in FAT to also perform the elimination processing and the write-in processing on a flash memory in a memory access driver.

[0012] Moreover, although the processing which opens a file and usually writes in data in the case of data writing is needed as a file system when FAT is being used As internal processing of the equipment in this case Modification of data made by writing in the data after changing into the block of a predetermined physical address [finishing / elimination] in the elimination unit of a flash memory, and changing into the logical address before changing the logical address, or the swap processing of a block which eliminates a part, And there was un-arranging [of having required the processing time for performing complicated processing of the processing which

changes the logical address which a FAT filesystem uses, and the physical address which is the order of a physical list of a flash memory with a translation table]. The invention in this application does not need an external device according to such a request, but it is the the best for personal digital assistant equipment, and aims at building the storage system which can realize efficient processing.

[0013]

[Means for Solving the Problem] Updating data for the invention in this application to update the data memorized inside data rewriting equipment, In data rewriting ~~equipment with the removable record medium with which the discernment data which~~ identify that program data, and the updating data and program data for transmitting updating data are recorded on data rewriting equipment were recorded The 1st storage means which the program data for transmitting updating data are read into the data rewriting equipment recorded on the record medium, and is remembered to be an insertion means to insert a removable record medium in the body of equipment, A detection means to detect discernment data from a record medium, and 2nd storage means by which are in the interior of rewriting equipment and the data updated are memorized, When discernment data are detected by the detection means from a record medium After reading the program data for transmitting updating data to data rewriting equipment based on the detected discernment data from the record medium and making the 1st storage means memorize, It aims at offering the data rewriting equipment characterized by having the control means which updates the data which read the updating data recorded on the record medium based on the program data memorized by the 1st storage means, and are memorized by the 2nd storage means inside data rewriting equipment.

[0014] Furthermore, updating data for the invention in this application to update the data memorized inside data rewriting equipment, The program data for transmitting updating data to data rewriting equipment, In the control approach of transmitting and storing updating data in data rewriting equipment from the body of data rewriting equipment with which the discernment data which identify updating data and program data being recorded were recorded, and a removable record medium The detection step which detects discernment data from a record medium, and when discernment data are detected from a record medium The read-out step which reads program data from a record medium based on the detected discernment data, The updating data for updating the data memorized inside data rewriting equipment from the record medium based on the read program data are read. It aims at offering the control approach characterized by having the updating step which updates the data memorized inside data rewriting equipment by the read updating data.

[0015] Furthermore, the invention in this application aims at providing the record medium which it had in the updating data storage area where the updating data for updating the data memorized inside the data rewriting equipment with which it is

equipped are recorded, the program data storage area where the program data for transmitting updating data are recorded on data rewriting equipment, and the discernment data storage area where the discernment data which identify that updating data and program data are recorded are recorded.

[0016] According to such invention-in-this-application equipment, the following operations are carried out. A control means is beforehand controlled in the rewritable condition, and the function for making rewriting possible was prepared into the rewriting program memorized to the record medium. This rewriting program is read into a control means, and the rewriting prohibition condition of a control means is canceled for the first time, it is controlled by the rewritable condition, and rewriting of program data can be performed by writing new program data in a control means by this rewriting program.

[0017] Namely, in this invention, while taking into consideration the safety for preventing rewriting or elimination of program data depended carelessly, actuation of the microcomputer as a control means of a Personal Digital Assistant is changed into a stable condition.

[0018]

[Embodiment of the Invention] Hereafter, the gestalt of operation of this invention is explained. In addition, the storage as a gestalt of operation shall be used as the tabular memory which has a tabular appearance configuration, and the personal digital assistant with which the Personal Digital Assistant equipment of this invention can perform record playback of a file to tabular memory shall correspond.

[0019] [the appearance configuration of tabular memory] -- the appearance configuration of the tabular memory which is the storage of this example is explained first. Tabular memory equips the interior of a tabular case with the memory device of for example, predetermined capacity. As this example, a flash memory (Flash Memory) is used as this memory device.

[0020] The terminal area which is missing from a base side from the transverse-plane lower part of a case, for example, has nine electrodes is formed, and internal read-out to a memory device or write-in actuation is performed from this terminal area through the attachment-and-detachment device by the side of the body of a personal digital assistant mentioned later. Let the upper left section of the direction of a flat surface of a case be a notch. In case this notch loads the attachment-and-detachment device for example, by the side of the body of a personal digital assistant with this tabular memory, it becomes a thing for preventing mistaking the path of insertion. Furthermore, the slide switch to prevent incorrect elimination of the contents of storage is formed in the base side.

[0021] [A format of tabular memory]

The format in a [memory file system processing hierarchy], then the system which uses tabular memory as a storage is explained. Drawing 1 shows the file system

processing hierarchy of the system which uses tabular memory as a storage. As shown in drawing 1 , as a file system processing hierarchy, the file management processing layer 61, the logical address layer 62, the physical address layer 63, and the flash plate memory access layer 64 set one by one under the application process layer 60. On this hierarchy, the file management processing layer 61 serves as the so-called FAT (File Allocation Table). Moreover, although the concept of the logical address and a physical address is introduced in the file system of this example as shown in this drawing, about this, it mentions later.

[0022] With the gestalt of this operation, it sets in the application process layer 60 especially. By the exclusive device By performing alter operation of processing of the address generation of the ROM data file which memorized the programme loader file which memorized the rewriting program which performs rewriting of the program data mentioned later, and program data The address file processing section 65 which starts processing of an address file is formed. In this case With reference to the physical address table 67 immediately placed, for example into RAM, the physical address on the flash memory corresponding to the logical address of a predetermined index file is acquired. The writing of a physical address is performed by the physical address write-in processing section 66 prepared in the flash plate memory access layer 64. Then, in the information terminal unit later mentioned to the tabular memory taken out from the exclusive device, write-in processing of a ROM data file is performed using a programme loader file.

[0023] Thus, in address file processing, since the flash plate memory access layer 64 can be performed immediately, without using the file management processing layer 61, the logical address layer 62, and the physical address layer 63, in spite of using the file in a FAT filesystem, complicated processing of a FAT filesystem is not needed but the processing time can be shortened. In addition, the exclusive device mentioned above means what has the function which writes a physical address in tabular memory with reference to the physical address table 67 which consists of RAM using the file system which performs address administration of the logical addresses, such as a FAT filesystem, and a physical address.

[0024] Especially with the gestalt of this operation, the physical address table 67 is formed in an exclusive device. By the address file processing section 65 which starts the address file processing realized as a function of a microcomputer by pressing the key which starts processing of the address file of a control unit With reference to the physical address table 67, the physical address on the flash memory corresponding to the logical address of a predetermined programme loader file and a ROM data file is acquired immediately. The writing of the physical address of a programme loader file and a ROM data file is performed by accessing to the tabular memory 7 in which the flash memory was built by the physical address write-in processing section 66 realized as a function of a microcomputer.

[0025] In addition, about record or regeneration of a record data file or a playback data file, the application process layer 60, the file management processing layer 61, the logical address layer 62, the physical address layer 63, and the flash plate memory access layer 64 are used altogether.

[0026] The physical DS of a flash memory which is a storage element in [physical DS] tabular memory is explained. A fixed-length data unit called a segment in the storage region as a flash memory serves as Daigen. This segment is size specified as 4MB per one segment, or 8MB, and the numbers of segments in one flash memory differ depending on the capacity of that flash memory.

[0027] And 8KB or 16KB divide this one segment as a fixed-length data unit called a block. In principle, since one segment is divided into 512 blocks, it will be set to $n = 511$ about Block n . However, in a flash memory, since the block count as defect area which is damage area [that it cannot write in] is permitted in the range of a predetermined number, an object, then Above n become less than 511 the substantial block count by which data writing is confirmed.

[0028] Two top blocks 0 and 1 are called boot block among the block 0 formed - n . However, two blocks are to be specified as a boot block from the head of an effective block in fact, and there is no guarantee whose boot blocks are not necessarily blocks 0 and 1. And the remaining block turns into a user block with which user data are stored.

[0029] 1 block is divided by a page 0 - m . Capacity of 1 page is made into 528 (= 512+16) cutting tools' fixed length who consists of 512 bytes of data area, and 16 bytes of redundancy section. In addition, about the structure of the redundancy section, it mentions later. Moreover, as pagination in 1 block, when the capacity of 1 block is 8KB, in the case of 16 pages and 16KB, it becomes 32 pages.

[0030] Such page structure within a block is common to the above-mentioned boot block and a user block. Moreover, read-out and the writing of data shall be performed per page, and it shall be carried out in a flash memory by elimination of data per block. And it shall not be carried out by the writing of data only to a page [finishing / elimination]. Therefore, actual rewriting and the actual writing of data will be performed for a block unit.

[0031] As for a top boot block, the information with which a header is stored and indicates the address to be to a page 1 as a location of initial failure data is stored to a page 0. Moreover, the information called CIS/IDI (Card Information Structure/Identify Drive Information) is stored in a page 2. Let the 2nd boot block be a field for the backup as a boot block.

[0032] After being based on [the concept of a physical address and the logical address], next physical DS of a flash memory which was mentioned above, according to data rewriting actuation, the concept of the physical address in the file system of this example and the logical address is explained.

[0033] A physical address is attached to each block. This physical address is decided according to the physical order of an array of the block in memory, and the relation with the physical address matched with a certain block and this becomes eternal.

[0034] Here, there are block used with which block data is memorized, and an intact block which block data was eliminated, i.e., is a non-record section.

[0035] And let the logical address be the address assigned as accompanies the data written in to the block. And it considers as the address which the FAT filesystem which this logical address mentions later uses.

[0036] Here, suppose that the contents rewrite or a part is eliminated from the above-mentioned condition as renewal of the data stored in the physical address. In such a case, in the file system of a flash memory, the updated data is written in writing in again the data updated to the same block to an intact block, without carrying out.

[0037] And in the condition before renewal of data, a change about the logical address is made so that the logical address corresponding to the physical address before updating may be equivalent to the physical address after renewal of the block with which the updated data were written in.

[0038] That is, it can be concluded that a physical address is the address given to a proper to a block, and the logical address is the address around which it attaches and turns as accompanies the data once written in to the block and which becomes peculiar to the write-in data of a block unit.

[0039] Being repeatedly accessed intensively by swap processing of such a block being performed to the block which is a certain same storage region is lost, and it becomes possible to prolong the life of a flash memory with the upper limit of the count of rewriting.

[0040] And by treating the logical address as mentioned above in this case, even if it makes it there be migration of the block written in by swap processing of a block by the data updating before and after updating, from FAT, the same address can be seen and subsequent accesses can be performed proper.

[0041] In addition, swap processing of a block is specified as what is completed within 1 segment for the purpose of making simple management for updating on the logic-physical address translation table mentioned later etc. Conversely, if it says, if it is made for swap processing of a block to straddle between segments, it will not be performed.

[0042] Correspondence of a physical address and the logical address changes by swap processing of a block being performed so that the [logic-physical address translation table] above-mentioned explanation may show. Therefore, in order to realize access for the writing of data to a flash memory, and read-out, the logic-physical address translation table in which correspondence with a physical address and the logical address is shown is needed. That is, the physical address corresponding to the logical address specified by FAT is specified, and it becomes

possible to access the block shown by this specified physical address because FAT refers to a logic-physical address translation table. Conversely, if it says and there will be no logic-physical address translation table, access to the flash memory by FAT will become impossible.

[0043] The example of structure of a logic-physical address translation table is shown. A logic-physical address translation table is stored to a certain block with which it is specified that it is in the segment of the last of a flash memory. First, the 2-page field which consists of pages 0 and 1 among the pages into which a block is divided is assigned as a logic-physical address translation table for segment 0. For example, since it has only one segment if a flash memory is the capacity which is 4MB, the field of only these pages 0 and 1 turns into a field of a logic-physical address translation table.

[0044] Moreover, since it has two segments, for example if a flash memory is the capacity which is 8MB, in addition to the pages 0 and 1 assigned as a logic-physical address translation table for segment 0, it will be assigned as a logic-physical address translation table for [2 pages] segment 1 of the pages 2 and 3 following this.

[0045] Henceforth, according to the increment in the capacity of a flash memory, the quota field of the logic-physical address translation table for every segment will be set up every continuing 2 pages. And since 16 segments exist if it is the case where it has the maximum capacity of 128MB, the page to the object for segments 15 will be assigned as a field of a logic-physical address translation table at the maximum. Therefore, in a flash memory with a maximum capacity of 128MB, 30 pages will be used and it is set to $N = 29$ at the maximum as a page N . A logic-physical address translation table is managed for every segment so that old explanation may show.

[0046] The data area for 2 pages is extracted and shown as what shows the structure of the logic-physical address translation table per one segment. That is, since a 1-page data area is 512 bytes, 1024 (= 512x2) cutting tools are developed.

[0047] Here, the relation between flash memory capacity and the size of a logic-physical address translation table is explained. As explained, the size of the logic-physical address translation table for managing one segment becomes 1024 bytes for 2 pages, i.e., 1KB. Therefore, by the capacity of 4MB by which a flash memory is equivalent to one segment, a logic-physical address translation table serves as size of 1KB. Moreover, a logic-physical address translation table is set to 2KB equivalent to 4 pages in 8MB by which the capacity of a flash memory is equivalent to two segments.

[0048] Moreover, when the capacity of a flash memory is 16MB, at the thing of 2048 block = 4 segment, a logic-physical address translation table becomes 2KB equivalent to 4 pages by the thing of 4KB and 1024 block = 2 segment by which a logic-physical address translation table is equivalent to 8 pages.

[0049] And by 4KB to which a logic-physical address translation table is equivalent to

8 pages in 32MB by which the capacity of a flash memory is equivalent to four segments, and 64MB by which the capacity of a flash memory is equivalent to eight segments, a logic-physical address translation table is set to 8KB equivalent to 16 pages, and a logic-physical address translation table is set to 16KB equivalent to 32 pages by 128MB by which the capacity of a flash memory is equivalent to the 16 greatest segments.

[0050] Drawing 2 explains the configuration of the Personal Digital Assistant of this example, and tabular memory [configuration of a Personal Digital Assistant and tabular memory]. Continuously, Drawing 2 shows the configuration of the main frame which is the body of a set of the Personal Digital Assistant which can perform read-out, store, and edit of data corresponding to the tabular memory explained until now. A file memory system is constituted by the body 20 of a set and the tabular memory 7 which are shown in this drawing.

[0051] as [reproduce / in addition, / body / from audio data, such as a video data, still picture data, and voice data, CD (Compact Disc), MD (Mini Disc TM), etc. / the class of main data which can be treated as an object of the store to the tabular memory 7 or read-out has the various body 20 of a set, for example,] -- high -- there are tone quality audio data, data for control, etc.

[0052] Although this example explains the speech data as main data as a system memorized and reproduced for simplification of explanation, it cannot be overemphasized by having the I/O system and processors of data, such as an animation, a still picture, and an audio, in the body 20 of a set that it can do with the storage system of data files, such as these animations.

[0053] As a configuration of the body 20 of a set, it has the attachment-and-detachment device 13 in which it is loaded with the body 20 of a set concerned removable, and transfer of the data of the tabular memory 7 with which this attachment-and-detachment device 13 was loaded, and the inclusion mold microcomputer 1 for control is performed by minding the host interface IC which is not illustrated.

[0054] Moreover, this body 20 of a set is equipped with the microphone which is not illustrated, for example, and audio signals collected by this microphone, such as a speech and dictate, are supplied to DSP (Digital Signal Processor) through microphone amplifier as an audio signal. In DSP, this inputted audio signal is changed into digital audio data, it gives including necessary signal processing, such as encoding processing, and a microcomputer 1 is supplied as record data. A microcomputer 1 is made possible [performing processing for recording this record data on the tabular memory 7 through the host interface IC].

[0055] Moreover, a microcomputer 1 reads the audio data currently recorded on the tabular memory 7 through the host interface IC, and supplies this read data to DSP. In DSP, necessary signal processing including recovery processing is performed about

the supplied data, and, finally it outputs to loudspeaker amplifier as an analog audio signal. With loudspeaker amplifier, the inputted audio signal is amplified and it outputs to a loudspeaker. Thereby, a playback audio is outputted.

[0056] Moreover, a microcomputer 1 is controlling a display driver and it is made possible to a display to display a necessary image of it. For example, the menu for actuation of a user, a guide display, or the display of the file content memorized by the tabular memory 7 is performed. Moreover, if the image data of an animation or a still picture is recorded, for example to the tabular memory 7, this image data is read and it is made possible to make it also make it display on a display.

[0057] With the gestalt of this operation here a microcomputer The flash ROM 2 which memorizes the ROM data file which memorized the program data for control, The interior RAM 3 which memorizes the programme loader file which memorized the rewriting program which performs rewriting of control data and program data The rewriting function of the program data memorized by the flash ROM 2. The write enable terminal 4 controllable in a prohibition condition or the effective condition, It has the jumper 14 which connects an output terminal 5, and the write enable terminal 4 and an output terminal 5, and is used for transfer of the control signal of a prohibition condition or an authorized state in the rewriting function of program data, and is constituted. Moreover, the exterior RAM 6 which memorizes temporarily the programme loader file which memorized the rewriting program which performs rewriting of program data is connected to the microcomputer 1 outside.

[0058] Thus, the microcomputer 1 which has the rewriting function of a flash ROM 2 is equipped with the write enable terminal 4 which controls the propriety of rewriting of a flash ROM 2 as a function of hardware. And in order to control this write enable terminal 4, the jumper 14 connects, and the output terminal 5 of a microcomputer usually rewrites the write enable terminal 4, and is controlling in the prohibition condition. Here, the write enable terminal 4 is controlled in the rewritable condition through a jumper 14 from an output terminal 5, and the function for enabling rewriting of a flash ROM 2 does not exist in the ROM data of the flash ROM 2 by the side of the body 20 of a set in an initial state.

[0059] In consideration of the safety for preventing rewriting or elimination of program data depended carelessly, the write enable terminal 4 is controlled by the gestalt of this operation in the rewritable condition through the jumper 14 beforehand wired, for example with the print pattern of a printed circuit board from the output terminal 5, and the function for enabling rewriting of a flash ROM 2 prepared into the programme loader which consists of software modules memorized in the tabular memory 7. This programme loader is read into inside [RAM / 3] a microcomputer 1, the rewriting prohibition condition of the write enable terminal 4 is canceled for the first time, and it is controlled by the rewritable condition, and it is constituted so that rewriting of ROM data can be performed by writing a new ROM data file in a flash ROM 2 with this

programme loader.

[0060] Moreover, with the gestalt of this operation, the programme loader file 9 which memorized the rewriting program which performs rewriting of program data, and the ROM data file 11 which memorized the new program data for control are memorized at the tabular memory 7 side.

[0061] Moreover, with the gestalt of this operation, the tabular memory discernment file 8 for loaders which identifies whether the programme loader file 9 and the ROM data file 11 this tabular memory of whose is a program for program data rewriting of normal are memorized is formed in the tabular memory 7 now connected to the body 20 of a Personal Digital Assistant through the desorption device 13.

[0062] Thus, it can prevent ROM data being rewritten and destroyed by the actuation which is not expected [overrun / of the microcomputer 1 resulting from the power source of the body 20 of a Personal Digital Assistant becoming unstable] by forming the programme loader file 9 in the tabular memory [that the tabular memory discernment file 8 for loaders is formed in the tabular memory 7, and not the body 20 of a set but] 7 side.

[0063] Moreover, with the gestalt of this operation, the programme loader address file 10 which memorized the address of the programme loader file 9, and the ROM data-address file 12 which memorized the address of the ROM data file 11 are formed in the tabular memory 7.

[0064] Here, especially the writing of the physical address of the programme loader address file 10 and the ROM data-address file 12 is performed by [as being the following]. As explained in drawing 1 , in an exclusive device, the physical address translation table 67 is formed on RAM. By the address file processing section 65 realized as a function of the microcomputer which starts address file processing by pressing the key which starts processing of the address file of a control unit With reference to the physical address table 67, the physical address on the flash memory corresponding to the logical address of a predetermined programme loader file and a ROM data file is acquired immediately. The physical address of a programme loader file and a ROM data file is written in by accessing to the tabular memory 7 by the physical address write-in processing section 66 realized as a function of a microcomputer.

[0065] In addition, various keys for a user to perform various actuation to the body 20 of a set are prepared in the control unit which is not illustrated. A microcomputer 1 receives the command according to the actuation performed to this control unit, and performs necessary control processing according to a command. As contents of actuation, ROM data rewriting directions, file memory directions, file selection directions, file playback directions, edit directions, etc. are enabled.

[0066] In addition, the configuration of the body 20 of a set shown in drawing 2 is an example to the last, and is not limited to this. That is, as long as the configuration

which can transmit and receive data is taken corresponding to the tabular memory 7, you may consider as what type of electronic equipment.

[0067] Next, the outline of the usage actuation by drawing 3 is explained. In addition, about the same block as drawing 2, the same sign is attached among each part shown in drawing 3. The microcomputer 1 is equipped with a flash ROM 2 and RAM3, and in order to explain as functional block in drawing 3, a flash ROM 2 is divided into flash ROM A and flash ROM B, and it is drawn. However, it does not need to dissociate physically, and the same flash ROM is used, dividing it by the block address, and this can be realized.

[0068] The discernment file 8, the programme loader address file 10, the programme loader file 9, the ROM data-address file 12, and the ROM data file 11 are memorized by the tabular memory 7 like the block shown in drawing 2, respectively.

[0069] The control program before the addition of a function is memorized by the flash ROM of a microcomputer 1 as an initial state, and the data which the control program before a functional addition used are memorized by RMA3.

[0070] The rewriting programme loader 15 is memorized by flash ROM [of a microcomputer 1] A as a program for loading a programme loader to the programme loader address file list as distinction of whether the program memorized by the tabular memory 7 for rewriting the control program before the functional addition memorized by flash ROM B is memorized, and a program for rewriting.

[0071] While Personal Digital Assistant 20 is equipped with the tabular memory 7 the discernment file 8, the programme loader address file 10, the programme loader file 9, the ROM data-address file 12, and the ROM data file 11 are remembered to be, respectively, if directions of update of a program are inputted, the rewriting programme loader 15 memorized by flash ROM A will be performed. The rewriting programme loader 15 searches the discernment file 8, accessing the tabular memory 7 with a physical address, as shown in (1). It is judged that the rewriting program of the data for updating the control program memorized by the flash ROM prepared in the microcomputer 1 interior of Personal Digital Assistant 20 to the tabular memory 7 with which current Personal Digital Assistant 20 is equipped if the discernment file 8 is detected, and data is memorized.

[0072] If the discernment file 8 is detected, the rewriting programme loader 15 will access the interior of the tabular memory 7 with a physical address, as shown in (2), and will search the programme loader address file 10. If the programme loader address file 10 is detected, the rewriting programme loader 15 is loaded to RAM3 in which the programme loader 9 is formed inside the microcomputer 1 from the address which pinpointed the location where the programme loader 9 memorized by the tabular memory 7 based on the programme loader address file 10 as shown in (3) is memorized, and was specified as shown in (4).

[0073] After all the programme loaders 9 are loaded to RAM3, the rewriting

programme loader 15 passes control to the programme loader 16 developed on RAM3. A microcomputer 1 is made to execute the instruction jumped to the activation starting address of a programme loader 16 so that a program may be performed with a microcomputer 1 from the activation starting address of the programme loader 16 specifically developed on RAM3.

[0074] The programme loader 16 on RAM3 which acquired the right of execution control accesses the tabular memory 7 with a physical address, as shown in (5), and it searches a ROM address file. The programme loader 16 on RAM3 is written in flash ROM B, eliminating the field where the program to which flash ROM B of a microcomputer 1 is updated was recorded in the data 14 for rewriting which access the ROM data file 11 and are read from the tabular memory 7 according to the ROM data-address file 12 as shown in (7), if the ROM data-address file 12 is detected as shown in (6).

[0075] The programme loader 16 memorized by RAM3 if the required data of the ROM data files 11 memorized by the tabular memory 7 are altogether written in flash ROM B of a microcomputer 1 controls a microcomputer 1 so that the program which controls Personal Digital Assistant 20 which executed the RESET instruction and was recorded on the flash ROM 2 so that a microcomputer might carry out initialization actuation operates.

[0076] By performing an initialization program, a microcomputer 1 eliminates the programme loader 16 memorized on RAM3, and comes to return to the usual processing actuation.

[0077] Actuation of the gestalt of this operation is explained using [the flow chart of flash ROM rewriting actuation], next the flow chart of flash ROM rewriting actuation of drawing 4. In drawing 4, the case where ROM data are rewritten using the programme loader file 9 and the ROM data file 11 which were memorized by the tabular memory 7 to Personal Digital Assistant 20 controlled by the rewriting prohibition condition of the ROM data shown in drawing 2 is explained.

[0078] In drawing 4, flash ROM rewriting processing is started and a microcomputer 1 judges whether the tabular memory discernment file 8 for loaders is memorized by the tabular memory 7 connected to the body of a Personal Digital Assistant at step S1. Specifically, a microcomputer 1 detects whether the header as an identifier for identifying that this tabular memory has memorized the programme loader file 9 and the ROM data file 11 for program data rewriting of normal is prepared in the tabular memory discernment file 8 for loaders.

[0079] A physical address file configuration is shown in drawing 6. In drawing 6, access for the writing to the flash memory by the physical address mentioned above, read-out, or elimination is performed as follows. First, it writes in the 1st, and retrieval is performed in the order of a physical address from the head of a block until the right header 40 which performs access of read-out or elimination comes. If the right header

40 is searched with the 1st processing by the 2nd, the physical address 41 corresponding to this will be acquired. Thus, access is performed to a desired physical address.

[0080] The example of the header data of a discernment file is shown in drawing 7 A, drawing 7 B, and drawing 7 C. Tabular memory is connected to the case where a personal computer is connected to the body of a Personal Digital Assistant through an interface as the load approach of the programme loader file 9 and the ROM data file 11, and it loads from a personal computer, and the body of a Personal Digital Assistant, and it may load from tabular memory.

[0081] It sets to drawing 7 A and, in the case of loading from a personal computer, is ABC as discernment data 50. DEF MEMSTICK PROGRAM LOADER LOAD FROM PC is memorized and 32 is recorded as pagination 52 which shows the storage region per block following a tooth space 51.

[0082] It sets to drawing 7 B and, in the case of loading from tabular memory, is ABC as discernment data 53. DEF MEMSTICK PROGRAM LOADER LOAD FROM MS is memorized and 16 is recorded as pagination 55 which shows the storage region per block following a tooth space 54.

[0083] If it is recognized with a microcomputer 1 that it is the tabular memory on which the header data as such an identifier were detected and the ROM data of normal were recorded, it will progress to step S2.

[0084] Next, a microcomputer 1 reads the programme loader address file 10 which is an address file of a programme loader as a software module at step S2. As explained in drawing 1, in an exclusive device, the physical address table 67 is specifically formed on RAM. By the address file processing section 65 realized as a function of the microcomputer 1 which starts address file processing by pressing the key which starts processing of the address file of a control unit With reference to the physical address table 67, the physical address on the flash memory corresponding to the logical address of a predetermined programme loader file is acquired immediately. The physical address of a programme loader file is written in by accessing to the tabular memory 7 by the physical address write-in processing section 66 realized as a function of a microcomputer 1. By reading this programme loader address file 10, the need of the microcomputer 1 of Personal Digital Assistant 20 of interpreting large-scale programs, such as FAT, is lost.

[0085] At step S3, a microcomputer 1 reads the programme loader file 9 the programme loader was remembered to be as a software module on the interior RAM 3. Using the physical address data of the programme loader address file 10, a microcomputer 1 accesses the storage region of the tabular memory 7, reads the programme loader file 9 the programme loader was remembered to be as a software module, and, specifically, writes it in the predetermined field on the interior RAM 3.

[0086] The example of the header data of a programme loader file is shown in drawing

7 C. It sets to drawing 7 C and is ABC as discernment data 56. DEF MEMSTICK PROGRAM LOADER is memorized, romdata.dat is recorded as a file name 57, and 0x00 A3, 0x003F, and 0xFFFF are recorded as physical block-address data 59 following a tooth space 58.

[0087] By step S4, a microcomputer 1 reads the ROM data-address file 12 which is an address file of a ROM data file. As explained in drawing 1, in an exclusive device, the physical address translation table 67 is specifically formed on RAM. By the address file processing section 65 realized as a function of the microcomputer 1 which starts address file processing by pressing the key which starts processing of the address file of a control unit With reference to the physical address table 67 on RAM, the physical address on the flash memory corresponding to the logical address of a predetermined ROM data file is acquired immediately. The physical address of a ROM data-address file is written in by accessing to the tabular memory 7 by the physical address write-in processing section 66 realized as a function of a microcomputer 1. By reading this ROM data-address file 12, the need of the inclusion microcomputer 1 of Personal Digital Assistant 20 of interpreting large-scale programs, such as FAT, is lost.

[0088] In addition, since the programme loader file 9 and the programme loader address file 10 which were mentioned above are recorded on the FAT filesystem, they can be easily written in Personal Digital Assistant 20 from a personal computer. At step S5, a microcomputer 1 reads the ROM data file 11 ROM data were remembered to be on the exterior RAM 6. Using the physical address data of the ROM data file 11, a microcomputer 1 accesses the storage region of the tabular memory 7, reads the ROM data file 11 ROM data were remembered to be, and, specifically, writes it in the predetermined field on the exterior RAM 6.

[0089] The example of the header data of a ROM data file is shown in drawing 7 C. It sets to drawing 7 C and is ABC as discernment data 56. DEF MEMSTICK PROGRAM LOADER is memorized, romdata.dat is recorded as a file name 57, and 0x00 A3, 0x003F, and 0xFFFF are recorded as physical block-address data 59 following a tooth space 58.

[0090] At step S6, a microcomputer 1 starts the programme loader on Interior RAM. Specifically, a microcomputer 1 outputs the control signal for starting the programme loader stored in Interior RAM, controlling an output terminal 5, and controlling from an output terminal 5 in the rewritable condition to the write enable terminal 4 through a jumper 14.

[0091] At step S7, a microcomputer 1 changes the write enable terminal 4 into the condition which can be written in. Specifically, a microcomputer 1 controls an output terminal 5 so that the write enable terminal 4 will be in a rewritable condition based on the control signal outputted from an output terminal 5.

[0092] Thus, since he is trying to prepare the function for enabling rewriting of a flash ROM 2 into the programme loader which consists of software modules memorized in

the tabular memory 7, this programme loader can cancel the rewriting prohibition condition of the write enable terminal 4 only after it can be read into inside [RAM / 3] a microcomputer 1, and can control it in the rewritable condition.

[0093] At step S8, a microcomputer 1 writes in the ROM data file on Exterior RAM. Specifically, a microcomputer 1 writes the ROM data stored in the exterior RAM 6 in a flash ROM 2.

[0094] Thus, rewriting of ROM data is performed by writing a new ROM data file in a flash ROM 2 by the programme loader.

[0095] Actuation of the gestalt of this operation is explained using [the flow chart of required data origination actuation], next the flow chart of required data origination actuation of drawing 5. In drawing 5, the case where the file 8 for tabular memory discernment for loaders memorized by the tabular memory 7 shown in drawing 2, the programme loader file 9, the ROM data file 11, the programme loader address file 10, and the ROM data-address file 12 are created by external devices, such as an exclusive device or a personal computer, is explained.

[0096] In drawing 5, data origination processing is started and tabular memory is initialized by the exclusive device at step S11. It initializes so that tabular memory may be connected to the exclusive device which can be formatted into file systems, such as FAT, and unnecessary data etc. may specifically be eliminated.

[0097] Tabular memory is connected to a personal computer at step S12. Specifically, tabular memory is connected to a personal computer through a predetermined interface.

[0098] At step S13, a programme loader file is written in from a personal computer. Since the programme loader file specifically beforehand created on the personal computer on the FAT filesystem by formatting tabular memory into file systems, such as FAT, by the exclusive device is recordable, a programme loader file can be easily written in the tabular memory 7 from a personal computer.

[0099] The example of the header data of a programme loader file is shown in drawing 7 C. It sets to drawing 7 C and is ABC as discernment data 56. DEF MEMSTICK PROGRAM LOADER is memorized, romdata.dat is recorded as a file name 57, and 0x00 A3, 0x003F, and 0xFFFF are recorded as physical block-address data 59 following a tooth space 58.

[0100] At step S14, a ROM data file is written in from a personal computer. Since the ROM data file specifically beforehand created on the personal computer on the FAT filesystem by formatting tabular memory into file systems, such as FAT, by the exclusive device is recordable, a ROM data file can be easily written in the tabular memory 7 from a personal computer.

[0101] The example of the header data of a ROM data file is shown in drawing 7 C. It sets to drawing 7 C and is ABC as discernment data 56. DEF MEMSTICK PROGRAM LOADER is memorized, romdata.dat is recorded as a file name 57, and 0x00 A3,

0x003F, and 0xFFFF are recorded as physical block-address data 59 following a tooth space 58.

[0102] At step S15, tabular memory is removed from a personal computer. Specifically, it removes from a personal computer by removing tabular memory from a predetermined interface.

[0103] At step S16, the file for tabular memory discernment for loaders is created by the exclusive device. In the case of loading from a personal computer, it is ABC as discernment data 50 as header data of a discernment file specifically shown in the location of a header 40 in the physical address file configuration shown in drawing 6 for example, at drawing 7 A. DEF MEMSTICK PROGRAM LOADER LOADFROM PC is memorized and 32 is recorded as pagination 52 which shows the storage region per block following a tooth space 51. Moreover, it sets to drawing 7 B and, in the case of loading from tabular memory, is ABC as discernment data 53. DEF MEMSTICK PROGRAM LOADER LOAD FROM MS is memorized and 16 is recorded as pagination 55 which shows the storage region per block following a tooth space 54.

[0104] In step S17, the address file of a programme loader file is created by the exclusive device. As explained in drawing 1, in an exclusive device, the physical address table 67 is specifically formed in RAM. By the address file processing section 65 realized by the function of the microcomputer 1 which starts address file processing by pressing the key which starts processing of the address file of a control unit With reference to the physical address table 67, the physical address on the flash memory corresponding to the logical address of a predetermined programme loader file is acquired immediately. The physical address of a programme loader file is written in by accessing to the tabular memory 7 by the physical address write-in processing section 66 realized by the function of a microcomputer 1.

[0105] In step S18, the address file of a ROM data file is created by the exclusive device. As explained in drawing 1, in an exclusive device, the physical address table 67 is specifically formed in RAM. By the address file processing section 65 realized by the function of the microcomputer 1 which starts address file processing by pressing the key which starts processing of the address file of a control unit With reference to the physical address table 67, the physical address on the flash memory corresponding to the logical address of a predetermined ROM data file is acquired immediately. The physical address of a ROM data file is written in by accessing to the tabular memory 7 by the physical address write-in processing section 66 realized by the function of a microcomputer 1.

[0106] Next, the actuation which rewrites the program memorized by the flash ROM 2 built in the microcomputer 1 in drawing 2 using drawing 8 is explained. Drawing 8 attaches a physical block address and shows the data memorized by ROM of the tabular memory 7 interior.

[0107] The file for discernment of drawing 7 B is recorded on 0x0004 of the physical

block address of drawing 8 . This is a tabular memory discernment file for loaders memorized in the tabular memory 7 of drawing 2 .

[0108] The ROM data-address file of drawing 7 C is recorded on 0x0052 of the physical block address of drawing 8 . This is the ROM data-address file 12 memorized in the tabular memory 7 of drawing 2 .

[0109] The programme loader address file of drawing 7 D is recorded on 0x00A2 of the physical block address of drawing 8 . This is the programme loader address file 10 memorized in the tabular memory 7 of drawing 2 .

[0110] Moreover, 0x0041 of the physical block address of drawing 8 is the programme loader file 9 memorized in the tabular memory 7 of drawing 2 .

[0111] 0x00 A3 of the physical block address of drawing 8 and the data currently recorded on 0x003F are data which constitute the ROM data file 11 memorized in the tabular memory 7 of drawing 2 .

[0112] When rewriting the program memorized by the flash ROM 2 built in the microcomputer 1 prepared in Personal Digital Assistant 20, the program for which program rewriting first memorized by the flash ROM 2 is prepared is started.

[0113] By the physical block address, the program for which this program rewriting is prepared is directed sequentially from the first address, and accesses the inside of the tabular memory 7 shown in drawing 8 , and it distinguishes whether there is any header which shows that it is the tabular memory discernment file 8 for loaders as shown in the predetermined location of the data read from the block directed by the physical block address at drawing 7 B . When data are recorded like drawing 8 and 0x0004 of a physical block address is read, it can distinguish that the program and data for rewriting the program memorized by the flash ROM 2 built in the microcomputer 1 prepared in Personal Digital Assistant 20 are recorded on the tabular memory with which it was equipped. That is, it is ABC when the block of a physical block address of 0x0004 is read. DEF MEMSTICKPROGRAM LOADER LOAD FROM MS It can be judged that the programme loader file, the ROM data file, etc. are recorded on the tabular memory with which it is equipped by the ability of a character string to be checked.

[0114] If are recorded on the tabular memory 7 equipped with the tabular memory discernment file 8 for loaders and it will be distinguished, the programme loader address file 10 and the ROM data-address file 12 will be continuously detected out of tabular memory.

[0115] If the block is read sequentially from the physical block address 0x0000 and it is the programme loader address file 10, specifically, it is ABC. DEF MEMSTICK PROGRAM LOADER The block equivalent to drawing 7 D on which the character string of loader.dat is recorded is searched. Moreover, it is ABC if it is the ROM data-address file 12. DEF MEMSTICK PROGRAM LOADER romdata.dat The block equivalent to drawing 7 C on which the character string is recorded is searched.

[0116] When data are recorded like drawing 8 , a programme loader address file is detected in 0x0052 of a physical block address, and a ROM data-address file is detected in 0x00A2 of a physical block address.

[0117] If the programme loader address file 10 is detected, a microcomputer 1 will read the physical address with which the program which rewrites the flash ROM 2 currently recorded on the tabular memory 7 is recorded from the programme loader address file 10. If it is the case of drawing 8 , 0x0041 will be first read from the location of the physical block-address data 73 of drawing 7 D.

[0118] A microcomputer 1 reads a programme loader file from the block of the tabular memory 7 of 0x0041 based on the physical block address which becomes 0x0041, and the interior RAM 3 is made to memorize it. In order to distinguish whether the programme loader file following the block of 0x0041 is recorded on the tabular memory 7 from the programme loader address file 10, the following physical block-address data are read from the location of the physical block-address data 73 of drawing 7 D. It can distinguish that the data which are read to a degree in the case of this example do not have the programme loader file which follows the physical block address 0x0041 since it is 0xFFFF.

[0119] Next, a microcomputer 1 reads the physical block address 59 of drawing 7 C from the detected ROM data-address file, and 0x00 A3 is read as first physical block address. It is based on read physical block-address 0x00 A3, and a microcomputer 1 is 1st from the block of physical block-address 0x00 A3 of the tabular memory 7. ROM The exterior RAM 6 which read Data and was established in Personal Digital Assistant 20 is made to memorize. Then, a microcomputer 1 reads 0x003F from the physical block-address data 59 of the ROM data-address file of drawing 7 C as 2nd physical block address. It is based on 0x003F of the read physical block address, and microcomputers 1 are 2nd(s) from physical block-address 0x003F of the tabular memory 7. ROM 1 block of Data is read and the exterior RAM 6 is made to memorize. Furthermore, a microcomputer 1 gets to know that there are no ROM data more than this by reading the termination data which are 0xFFFF as a following physical block address from the physical block-address data 59 of the ROM data-address file of drawing 7 C, and ends read-out processing of ROM data.

[0120] If each file of the programme loader file 9 and the ROM data file 11 is memorized by each of the interior RAM 3 and the exterior RAM 6, the program for which rewriting memorized by the flash ROM 2 is prepared will be passed to the programme loader file remembered in the control that the programme loader file memorized to the interior RAM 3 operates to the interior RAM 3. It is carried out by jumping to the activation starting position of the programme loader specifically memorized to the interior RAM 3 being directed to a microcomputer 1.

[0121] The programme loader memorized to the interior RAM 3 which acquired the control of activation is programmed controllable in the output terminal 5 of a

microcomputer 1, and it controls an output terminal 5 so that the write enable terminal 4 will be in the condition which can rewrite a flash ROM 2 through a jumper 14. [0122] Then, the programme loader memorized to the interior RAM 3 eliminates the field where the data which replace the data memorized by the flash ROM memorized to the exterior RAM 6 at least are memorized, and writes the data memorized to the exterior RAM 6 in the predetermined block of a flash ROM 2. After writing is completed, an output terminal 5 is controlled, the write enable terminal 4 is set as the condition which cannot rewrite a flash ROM 2, and a boot loader ends rewriting processing of delivery and a flash ROM 2 for a control to the program memorized by the flash ROM 2 after that. It is carried out in jumping to the address with which activation is started by the initial state of the programs specifically memorized by the flash ROM 2.

[0123] The physical address acquisition procedure of the programme loader file of step S17 in the flow chart of drawing 5 mentioned above and step S18 and a ROM data file is explained using the file system processing hierarchy of drawing 1.

[0124] In drawing 1, a continuous cluster number is first acquired from the cluster number and FAT of a directory to a head of the application process layer 60 in the address file processing section 65 to the 1st. The cluster number obtained by the 1st processing is changed into the 2nd by referring to the physical address table 67 to a physical address. Thereby, the physical address of a desired programme loader file and a ROM data file is acquirable. Thus, the physical address is computed from the cluster number of the logical address.

[0125] As the file system hierarchy of [FAT structure] drawing 1 explained, file management processing will be performed by FAT. That is, with Personal Digital Assistant 20 of a configuration of having been shown in drawing 2, in order to realize record playback which is the data store and read-out to the tabular memory 7, the file memory location management by FAT will be referred to with the demand by the application process, logic-physical address conversion further mentioned above will be performed, and actual access will be performed. Here, the structure of FAT is explained.

[0126] The outline of the management structure by FAT is shown. In addition, in this example, although FAT and a logic-physical address translation table will be stored in the tabular memory 15, FAT structure turns into management structure within tabular memory.

[0127] And FAT management structure consists of the copy of a partition table, a free area, a boot sector, and FAT and FAT, a root directory, and a data area. a data area -- a cluster 2 and a cluster 3 -- although unit data are shown as ..., it becomes 1 data unit which this cluster is a management unit and is treated by FAT. Although cluster size is generally made into 4 K bytes as standard by FAT, this cluster size can take the magnitude of the exponentiation of two among 512 bytes - 32 K bytes.

[0128] Although one block is made into 8 K bytes or 16 K bytes by the tabular memory of this example as mentioned above, the cluster which treats by FAT in the case of the tabular memory 15 made into 1 block = 8 K bytes is made into 8 K bytes. Moreover, the cluster which treats by FAT in the case of the tabular memory 1 made into 1 block = 16 K bytes is made into 16 K bytes. That is, 8 K bytes or 16 K bytes are a data unit on FAT management, and it considers as one data unit as a block by tabular memory. Therefore, cluster size which will be treated by FAT if it sees from tabular memory = it becomes the block size of the tabular memory. For this reason, suppose that it thinks as a block = cluster for simplification about the explanation after this example.

[0129] and -- as a block number -- x -- although it was indicated as ... $(x+m-1)$, $(x+m)$, $(x+m+1)$, and ... $(x+m+2)$, the various data which build FAT structure, for example in each block in this way will be memorized. In addition, each information is not necessarily memorized in fact by each block which continues physically in this way.

[0130] In FAT structure, the head of a FAT partition and the address of termination which are made into a maximum of 2 G bytes are first described by the partition table. Magnitude, cluster size, the size of each field, etc. are described as the exception of the so-called 12bitFAT(s) and 16bitFAT, and FAT structure by the boot area.

[0131] A copy is described by the field which FAT serves as a table showing the link structure of the cluster which constitutes each file so that it may mention later, and continues about FAT. A file name, a head cluster number, and various attributes are described by the root directory. 32 bytes per file of these description is used.

[0132] [The file structure of tabular memory]

The file structure memorized by a [directory configuration], next tabular memory is explained. The example of a directory configuration is shown first. as mentioned above, as main data which can be treated by tabular memory, it is reproduced from CDs, MD, etc., such as audio data, such as a video data, still picture data, and a speech, and data for music, -- high, although there are tone quality audio data, data for control, etc. As directory structure, from a root directory to for this reason, VOICE (the directory for speeches) DCIM (directory for still pictures), MOxxxxnn (directory for animations), AVCTL (directory for control), and HIFI (directory for music) are allotted.

[0133] Suppose that the file of being audio data (speech data) is described in detail especially in this example. As a subdirectory of Directory VOICE, an order file (ORDER.MSF), an additional information management file (INFO.MSF), a folder (FOLDER1, FOLDER2 ...), etc. are formed. Moreover, for example in a folder, the files (file name 98120100.MSV etc.) of the speech data which are actual audio data are formed.

[0134] The DS of a [speech data file] speech data file is shown. A speech data file is constituted including a format frame (FORMAT FRAME), the TOC frame (TOC FRAME), a title frame (TITLEFRAME), a manufacturer frame (MAKER FRAME), an

author frame (AUTHOR FRAME), a space frame (SPACE FRAME), and a data frame (DATA FRAME). However, a speech data file will be constituted by the format frame (FORMATFRAME) which it is arbitrary to prepare a title frame (TITLE FRAME), a manufacturer frame (MAKERFRAME), and an author frame (AUTHOR FRAME), and was fundamentally enclosed with the thick frame, the TOC frame (TOC FRAME), a space frame (SPACE FRAME), and the data frame (DATA FRAME).

[0135] A format frame serves as fundamental management information of the speech data file concerned, and the class of codec etc. is shown. The TOC frame serves as management information showing the array condition of each frame of the speech data file concerned. That is, the frame structure of a speech data file is identified by description of the TOC frame. A data frame serves as a field where actual speech data are stored. Although a space frame is mentioned later in detail, it is made into intact area and functions for a setup of a reserve field in case this is expansion of the TOC frame, and the playback improper area in a file.

[0136] In such each frame, two or more data frames and space frames may exist in the same file. However, a data frame and frames other than a space frame are set to one in the same file. Moreover, a format frame is surely arranged at the head of a file. And the TOC frame is arranged just behind a format frame. The title frame which is a frame which was mentioned above, and which is not indispensable, a manufacturer frame, and an author frame are collectively arranged after the TOC frame. These will be arranged at the sequence that Frame ID is small. Moreover, a space frame is surely arranged at the degree of the list of a frame which is not indispensable.

[0137] Moreover, let the data area which is not reproduced be a space frame in the cluster in which the division point is included at the time of DEBAIDO processing of a speech data file. That is, when the location in the middle of a certain cluster is made into a dividing point, in the file of the second half by division, the head of speech data serves as a dividing point of the cluster. To the last, although the data which must not reproduce the dividing point earlier data in the cluster will be contained in this way since the smallest unit of a file serves as a cluster, the part is made not to be reproduced as a space frame.

[0138] moreover, the result of division of a speech data file, and joint processing -- two space frames -- ** -- it can become the situation which becomes each other When two space frames which become each other such exist, it is summarized to one space frame. That is, it is made to specifically be managed with the TOC frame as one space frame.

[0139] As mentioned above, although the flash ROM rewriting actuation by the Personal Digital Assistant as a gestalt of operation and tabular memory, and these has been described, this invention is not limited to these configurations and actuation. Although rewriting processing of the ROM data for the writing of the audio data file whose main data are voice data as audio data, or a voice data file, read-out, and

elimination was especially mentioned as the example in the system using tabular memory in the gestalt of operation, as mentioned above, in the system using tabular memory, image data files, such as audio data files, such as a music data file, and an animation data file, etc. can be treated in addition to a voice data file. And actuation of the microcomputer of Personal Digital Assistant 20 can be stabilized by rewriting processing of ROM data being performed about this audio data file and an image data file as well as the above.

[0140] Thus, a FAT filesystem is used as a file system. Use a flash memory through a ~~desorption device as a record medium, and it has the function~~ which records or reproduces the stream data which are audio data, such as a speech and music. In the Personal Digital Assistant which has the microcomputer which can rewrite ROM data Without disassembling a microcomputer in any way, it can become possible to rewrite ROM data, stabilization of actuation of the microcomputer of Personal Digital Assistant 20 can be attained, and improvement in the operability at the time of renewal of ROM data and repair and maintainability can be aimed at.

[0141] In addition, in ****, that all it is a FAT or FAT filesystem top mean the file management system in the nonvolatile memory which constitutes tabular memory. moreover, as a record medium applied to the Personal Digital Assistant of the gestalt of this operation Not as the thing limited to tabular memory like drawing 2 but as a solid-state memory medium made into other appearance configurations for example, [memory module / a memory chip, a memory card,] Siemens, SanDisk, Hitachi, Motorola, NEC, MultiMediaCard (MMC) which Nokia advocates, Toshiba, Matsushita, SD memory card (SD[Secure Digital] Memory Card) that SanDisk advocates or a PC card, etc. may be used.

[0142] Moreover, a format of the file system explained until now is not limited to a FAT filesystem, either, other file systems may be used, and it actually responds, for example and a convention of the details etc. may be changed. Furthermore, it is not limited to what also mentioned the variation of flash memory capacity above. Of course, the memory device of the storage of the gestalt of this operation may not be restricted to a flash memory, but other memory devices and a rewritable compact disk are sufficient as it.

[0143] Moreover, all the functions for enabling rewriting of the flash ROM 2 which is the nonvolatile memory in this invention are not supplied from the tabular memory 7, but you may make it supply a part of function for enabling rewriting of a flash ROM 2 from the tabular memory 7.

[0144]

[Effect of the Invention] The record medium which memorized the rewriting program which the Personal Digital Assistant equipment of this invention is formed possible [desorption] to a body, and performs rewriting of program data, Since it had a discharge means to control to rewrite at the time of wearing of a record medium, to

rewrite to a control means based on a program, and to cancel the rewriting prohibition condition of a prohibition means once [at least] The function for enabling rewriting of program data By [which were memorized to the record medium] rewriting and preparing into a program, this rewriting program is read into a control means. The rewriting prohibition condition of program data is canceled for the first time, and it is controlled by the rewritable condition. By this rewriting program Since rewriting of program data can be performed by writing new program data in the interior of a body by the control means, It becomes possible [without disassembling the microcomputer as a control means in any way] to rewrite program data. The effectiveness that stabilization of actuation of the microcomputer of a Personal Digital Assistant can be attained, and improvement in the operability at the time of renewal of program data and repair and maintainability can be aimed at is done so.

DESCRIPTION OF DRAWINGS

[Brief Description of the Drawings]

[Drawing 1] It is the explanatory view of the file system processing hierarchy of the gestalt of operation.

[Drawing 2] It is drawing showing the configuration of the Personal Digital Assistant of the gestalt of operation, and tabular memory.

[Drawing 3] It is drawing showing the microcomputer of the gestalt of operation, and the data flow between tabular memory.

[Drawing 4] It is the flow chart which shows flash ROM rewriting actuation of the gestalt of operation.

[Drawing 5] It is the flow chart which shows creation actuation of the required data of the gestalt of operation.

[Drawing 6] It is drawing showing the configuration of the physical address file of the gestalt of operation.

[Drawing 7] It is drawing showing the example of the header data of the gestalt of operation, and drawing 7 A is the header data of the file for discernment, in the case of loading from a personal computer, drawing 7 B is the header data of the file for discernment, and, in the case of loading from tabular memory, drawing 7 C and drawing 7 D are the header data of a programme loader file and a ROM data file.

[Drawing 8] It is drawing showing the data storage condition inside the tabular memory the program which rewrites the flash ROM of the gestalt of operation was remembered to be.

[Description of Notations]

1 A microcomputer, 2 .. A flash ROM, 3 .. Interior RAM 4 A write enable terminal,

5 .. An output terminal, 6 .. Exterior RAM 7 Tabular memory, 8 .. The file for tabular memory discernment for loaders, 9 .. Programme loader file, 10 A programme loader address file, 11 .. ROM address file, 12 A ROM data-address file, 13 .. A desorption device, 14 .. Jumper, 20 A Personal Digital Assistant (body of a set), 40 .. A header, 41 .. Physical address, 50 Discernment data, 51 .. A tooth space, pagination per 52..block, 53 Discernment data, 54 .. A tooth space, pagination per 55..block, 56 [.. Physical block-address data, 60 / .. An application process layer, 65 / .. The address file processing section, 67 / .. A physical address table (RAM) 56 / .. Physical address write-in processing section, -] Discernment data, 57 A file name, 58 .. A tooth space, 59

(19) 日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11) 特許出願公開番号
特開2001-188686
(P2001-188686A)

(43) 公開日 平成13年7月10日 (2001.7.10)

(51) Int.Cl. ⁷	識別記号	F I	テ-マコ-ド [*] (参考)
G 0 6 F 11/00		G 0 6 F 12/00	5 5 0 K
12/00	5 5 0	12/14	3 1 0 F
12/14	3 1 0	15/02	3 3 5 G
15/02	3 3 5	G 0 6 K 17/00	D
G 0 6 K 17/00		G 0 6 F 9/06	6 3 0 A

審査請求 未請求 請求項の数20 O L (全 19 頁) 最終頁に続く

(21) 出願番号 特願2000-294387(P2000-294387)

(22) 出願日 平成12年9月27日 (2000.9.27)

(31) 優先権主張番号 特願平11-301335

(32) 優先日 平成11年10月22日 (1999.10.22)

(33) 優先権主張国 日本 (J P)

(71) 出願人 000002185

ソニー株式会社

東京都品川区北品川6丁目7番35号

(72) 発明者 飯田 健一

東京都品川区北品川6丁目7番35号 ソニ
ー株式会社内

(72) 発明者 野口 真理

東京都品川区北品川6丁目7番35号 ソニ
ー株式会社内

(72) 発明者 大林 秀次

東京都品川区北品川6丁目7番35号 ソニ
ー株式会社内

(74) 代理人 100080883

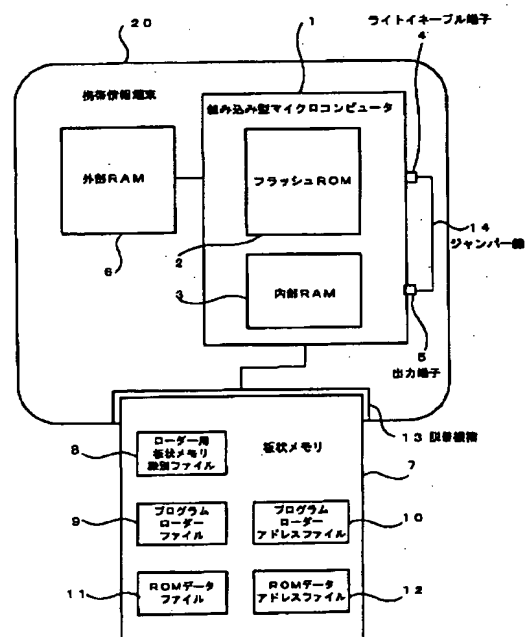
弁理士 松隈 秀盛

(54) 【発明の名称】 データ書換装置、制御方法および記録媒体

(57) 【要約】

【課題】 外部装置を必要とせず、携帯端末装置に最適で、効率的な処理を実現できるデータ書換装置、制御方法および記録媒体を提供することを課題とする。

【解決手段】 携帯情報端末装置本体20に備えられた書き換えの制限された不揮発性メモリであるフラッシュROM2に記憶されたデータを、着脱可能な記録媒体である板状メモリ7から供給される更新を実行するプログラムと更新に使用されるデータとを利用して行うものである。



携帯情報端末および板状メモリの構成を示す図

【特許請求の範囲】

【請求項1】 データ書換装置内部に記憶されているデータを更新するための更新データと、上記データ書換装置に上記更新データを転送するためのプログラムデータと、上記更新データ及びプログラムデータが記録されていることを識別する識別データが記録された記録媒体が着脱可能なデータ書換装置において、上記着脱可能な記録媒体を装置本体に挿入する挿入手段と、

上記記録媒体に記録された上記データ書換装置に上記更新データを転送するためのプログラムデータが読み込まれて記憶される第1の記憶手段と、

上記記録媒体から上記識別データを検出する検出手段と、

上記書換装置内部にあり、上記更新されるデータが記憶されている第2の記憶手段と、

上記検出手段にて上記記録媒体から上記識別データが検出された場合には、上記検出された識別データに基づいて上記データ書換装置に上記更新データを転送するためのプログラムデータを上記記録媒体から読み出して上記第1の記憶手段に記憶させた後、上記第1の記憶手段に記憶されたプログラムデータに基づいて上記記録媒体に記録された上記更新データを読み出して上記データ書換装置内部の第2の記憶手段に記憶されているデータを更新する制御手段とを備えたことを特徴とするデータ書換装置。

【請求項2】 請求項1記載のデータ書換装置において、上記更新データは、上記データ書換装置を動作させるためのプログラムデータであることを特徴とするデータ書換装置。

【請求項3】 請求項1記載のデータ書換装置において、上記データ書換装置は、上記記録媒体に記録されている上記更新データを転送するためのプログラムデータを上記第1の記憶手段に転送して記憶させるロードプログラムデータが記憶された第3の記憶手段を更に備えることを特徴とするデータ書換装置。

【請求項4】 請求項1記載のデータ書換装置において、上記データ書換装置は、上記データ書換装置内部に記憶されているデータの書き換えを規制する規制手段を更に備え、上記記憶手段に記憶されたプログラムデータによって書き換えの規制を解除することを特徴とするデータ書換装置。

【請求項5】 請求項2記載のデータ書換装置において、上記データ書換装置に上記更新データを転送するための

プログラムデータを読み込んだ後に上記書き換えの規制の解除が可能となることを特徴とするデータ書換装置。

【請求項6】 請求項1記載のデータ書換装置において、上記データ書換装置内部に記憶されているデータは不揮発性メモリ手段に記憶されていることを特徴とするデータ書換装置。

【請求項7】 請求項1記載のデータ書換装置において、

上記データ書換装置は、

第4の記憶手段を更に備え、

上記データ書換装置内部に記憶されているデータを更新するための更新データは、上記第4の記憶手段に記憶された後に、上記データ書換装置内部の第2の記憶手段に記憶されているデータを更新するために使用されることを特徴とするデータ書換装置。

【請求項8】 請求項1記載のデータ書換装置において、

上記制御手段は、

上記データ書換装置に上記更新データを転送するためのプログラムデータが上記記録媒体に記録されている位置を示す第1の位置データを検索し、上記検索された第1の位置データに基づいて上記プログラムデータを読み出すことを特徴とするデータ書換装置。

【請求項9】 請求項1記載のデータ書換装置において、

上記制御手段は、

上記データ書換装置内部に記憶されているデータを更新するための更新データが上記記録媒体に記録されている位置を示す第2の位置データを検索し、上記検索された第2の位置データに基づいて上記記録媒体に記録された上記更新データを読み出すことを特徴とするデータ書換装置。

【請求項10】 データ書換装置内部に記憶されているデータを更新するための更新データと、上記データ書換装置に上記更新データを転送するためのプログラムデータと、上記更新データ及びプログラムデータが記録されていることを識別する識別データとが記録された上記データ書換装置本体と着脱可能な記録媒体から、上記更新データを上記データ書換装置に転送して記憶させる制御方法において、

上記記録媒体から上記識別データを検出する検出ステップと、

上記記録媒体から識別データが検出された場合には、上記検出された識別データに基づいて上記プログラムデータを上記記録媒体から読み出す読み出しステップと、上記読み出されたプログラムデータに基づいて上記記録媒体から上記データ書換装置内部に記憶されているデータを更新するための更新データを読み出して上記読み出された更新データで上記データ書換装置内部に記憶され

ているデータを更新する更新ステップとを備えたことを特徴とする制御方法。

【請求項11】 請求項10記載の制御方法において、上記データ書換装置が備える一時記憶手段に上記プログラムデータを記憶させることを特徴とする制御方法。

【請求項12】 請求項10記載の制御方法において、上記制御方法は、上記データ書換装置が備える一時記憶手段に上記記録媒体に記録された更新データを一時的に記憶させる一時記憶ステップを更に備えることを特徴とする制御方法。

【請求項13】 請求項10記載の制御方法において、上記制御方法の上記識別データを検出する検出ステップは、

上記記録媒体を構成する所定のデータサイズのブロックごとに所定のデータパターンが有るか否かで上記識別データを検出することを特徴とする制御方法。

【請求項14】 請求項10記載の制御方法において、上記制御方法のデータを更新する更新ステップは、上記データ書換装置に記憶された更新される位置のデータを消去してからデータを更新することを特徴とする制御方法。

【請求項15】 請求項10記載の制御方法において、上記制御方法は、データ書換装置内部に記憶されているデータを更新するための更新データを上記記録媒体から読み出すステップに先立ち、

上記更新データが上記記録媒体に記録されている位置を示す第1の位置データを検索する検索ステップを更に備え、

上記第1の位置データに基づいてデータ書換装置内部に記憶されているデータを更新するための更新データを上記記録媒体から読み出すことを特徴とする制御方法。

【請求項16】 請求項10記載の制御方法において、上記制御方法は、上記データ書換装置に上記更新データを転送するためのプログラムデータが上記記録媒体に記録されている位置を示す第2の位置データを検索する検索ステップを更に備え、

上記プログラムデータの読み出しは、上記第2の位置データに基づいて行われることを特徴とする制御方法。

【請求項17】 装着されるデータ書換装置内部に記憶されているデータを更新するための更新データが記録される更新データ記録領域と、

上記データ書換装置に上記更新データを転送するためのプログラムデータが記録されるプログラムデータ記録領域と、

上記更新データ及びプログラムデータが記録されていることを識別する識別データが記録される識別データ記録領域と、を備えた記録媒体。

【請求項18】 請求項17記載の記録媒体において、上記更新データが上記記録媒体に記録されている位置を

示す第1の位置データが記録される第1の位置記録領域を更に備えることを特徴とする記録媒体。

【請求項19】 請求項17記載の記録媒体において、上記プログラムデータが上記記録媒体に記録されている位置を示す第2の位置データが記録される第2の位置記録領域を更に備えることを特徴とする記録媒体。

【請求項20】 請求項18記載の記録媒体において、上記記録媒体は、

上記第1の位置データが記録される第1の位置記録領域とは別に、上記記録媒体に記録されたデータを管理する管理データが記録される管理データ記録領域を備えることを特徴とする記録媒体。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】 本願発明は、装置に備えられた書き換え制限がされた不揮発性メモリに記録されたデータの更新を、上記装置に着脱可能な記録媒体から供給される更新用のプログラムと更新データとに基づいて行うデータ書換装置、制御方法および記録媒体に関するものである。

【0002】

【従来の技術】 現在、携帯情報端末装置内部に組み込まれたマイクロコンピュータにおいては、プログラムを格納するROMデータ格納エリアをフラッシュメモリとして、基板に実装後もたとえば装置の機能を追加したり性能の向上を目的としてフラッシュメモリに記憶された装置を制御するためのプログラムやプログラムの動作時に必要となるデータを外部から書き換えることのできる機能を有しているものがある。通常これらのマイクロコンピュータに対しては、機能の一部として備えるシリアルインターフェースや装置外部からの書換を実現するためにシリアル通信インターフェースを装置に設けて、シリアル通信を用いて外部の書き込み装置からROMデータを書き換えることができるようにしている。

【0003】 一般に、これら機能追加や機能向上のためのROMデータの書き換え動作は、その装置の動作を決定する非常に危険な作業であるため、間違っただけで書き込み状態に入ってはならないように構成されている。従って、従来は、例えばサービスセンターや修理部門などでのROMデータ書き換え時には、装置の外筐を開けて装置内部の各部品にアクセス可能な状態にしておいてから書換動作に入るために必要な制御信号を伝達するためのハード的な結線を例えばジャンパー線をハンダ付けしたり上記結線の切り換えやスイッチの切り換え等の物理的な作業により行うようにすることにより、装置に記憶されているROMデータの書き換えを行うプログラムをユーザーによる誤操作などによっても誤ってROMデータの書き換え動作に入らないようにしていた。

【0004】 また、自動車用のナビゲーション装置では、ROMデータの書き換え用のCD-ROMなどをナ

ナビゲーション装置に装着して、電源の安定しているナビゲーション装置内に設けられたプログラムローダーによりROMデータの書き換えを行うことで新機能の追加やナビゲーションに使用するデータの更新への対応を行うようにしていた。

【0005】また、インターネットのルーター装置では、例えば新たなインターネットプロトコルやネットワークアプリケーションプログラムに対応するための、あるいはクラッキング等からLAN(Local Area Network)を守るためのプログラムを新たに追加したりするためにルーター装置内部に持つアップデート用のファイルをダウンロードするプログラムによってサーバーからROMデータの書き換えファイルがルーター装置に供給され、同様に、電源の安定しているルーター装置内に設けられたプログラムローダーによりROMデータの書き換え動作を行うようにしていた。

【0006】さらに、近年、例えばフラッシュメモリなどの固体記憶素子を搭載した小型の記憶媒体を形成し、専用のドライブ装置や、或いはドライブ装置をオーディオ/ビデオ機器、情報機器などに内蔵して、コンピュータデータ、画像データ、オーディオデータなどを記憶できるようにするものが開発されている。

【0007】

【発明が解決しようとする課題】ところが、上述したROMデータの書き換え作業は、誤操作などから誤ってROMデータが書き換わることがないようにするための装置の電気的回路の切り換えを行うために、サービスマンやサービスプロバイダなどで装置を分解して行ったり、ユーザー自らがプログラムの書き換えを行う場合であっても専用の治具を用意して装置の所定の外部端子に接続したりする必要があるという不都合があった。

【0008】また、電源の安定していない携帯端末装置に対しては、電源電圧の急激な変動や低下等によってマイクロコンピュータの動作が不安定となり、最悪の場合には暴走によってROMデータの書き換えを行うプログラムが動作してしまう恐れがあるため、ROMデータの書き換えのためのプログラムローダーを装置内に設けることができないという不都合があった。

【0009】また、このような固体記憶素子を利用した記憶システムにおいては、記憶したファイルに対する処理を効率よく行うことが求められている。例えば記憶媒体内でのファイル処理のために必要となるデータ移動・複製・書換などが最小限であることや、処理時間や消費電力を最小限とすることが要求されている。

【0010】これに対して、多くの上述したROMデータ等のプログラムの開発環境は、ファイルシステムとしてFAT(File Allocation Table)を使用しているため、記録媒体を介してROMデータをマイクロコンピュータに書き込む際にもROMデータがFAT等のファイルシステムとして記述されている

とROMデータのコピーに便利である。しかしながら、この場合、装置側では、FAT等の比較的大規模のファイルシステムを解釈する必要があるため、記録データの処理が非常に複雑になる。

【0011】具体的には、フラッシュメモリへの物理的アクセスに使用するための物理アドレスと、FATを用いてデータを論理空間で処理するための論理アドレスの間で、アドレスの変換作業が例えばフラッシュメモリ上を持つ論理空間で扱われる論理アドレスとフラッシュメモリを実際にアクセスする場合に使用する物理アドレスとを変換するための論理物理アドレス変換テーブルを使って装置内のメモリアクセスドライバ部において必要になる。また、フラッシュメモリの消去単位が書き込み単位よりも大きいという物理的仕様を前提としてこれを効率よく処理する手段が必要とされている。さらに、FATでは、データが上書き可能である記録媒体を前提にデータアクセスが行われるため、消去してから書き込みを行う必要のあるフラッシュメモリを使用する際には、FATでの書き込み要求に対して、フラッシュメモリ上での消去処理および書き込み処理もメモリアクセスドライバにおいて行う必要がある。

【0012】また、ファイルシステムとして、FATを使用している場合、通常はデータ書き込みの際に、ファイルをオープンしてデータを書き込んでいく処理が必要になるが、この場合の装置の内部処理としては、フラッシュメモリの消去単位で消去済みの所定の物理アドレスのブロックに変更後のデータを書き込んで論理アドレスを変更前の論理アドレスに変更することにより行われるデータの変更または一部消去を行うブロックのスワップ処理、およびFATファイルシステムが使用する論理アドレスとフラッシュメモリの物理的並び順である物理アドレスとを変換テーブルにより変換する処理などの複雑な処理を行うための処理時間を要するという不都合があった。本願発明はこのような要望に応じて、外部装置を必要とせず、携帯端末装置に最適で、効率的な処理を実現できる記憶システムを構築することを目的とする。

【0013】

【課題を解決するための手段】本願発明は、データ書換装置内部に記憶されているデータを更新するための更新データと、データ書換装置に更新データを転送するためのプログラムデータと、更新データ及びプログラムデータが記録されていることを識別する識別データが記録された記録媒体が着脱可能なデータ書換装置において、着脱可能な記録媒体を装置本体に挿入する挿入手段と、記録媒体に記録されたデータ書換装置に更新データを転送するためのプログラムデータが読み込まれて記憶される第1の記憶手段と、記録媒体から識別データを検出する検出手段と、書換装置内部にあり、更新されるデータが記憶されている第2の記憶手段と、検出手段にて記録媒体から識別データが検出された場合には、検出された識

別データに基づいてデータ書換装置に更新データを転送するためのプログラムデータを記録媒体から読み出して第1の記憶手段に記憶させた後、第1の記憶手段に記憶されたプログラムデータに基づいて記録媒体に記録された更新データを読み出してデータ書換装置内部の第2の記憶手段に記憶されているデータを更新する制御手段とを備えたことを特徴とするデータ書換装置を提供することを目的としている。

【0014】更に本願発明は、データ書換装置内部に記憶されているデータを更新するための更新データと、データ書換装置に更新データを転送するためのプログラムデータと、更新データ及びプログラムデータが記録されていることを識別する識別データとが記録されたデータ書換装置本体と着脱可能な記録媒体から、更新データをデータ書換装置に転送して記憶させる制御方法において、記録媒体から識別データを検出する検出ステップと、記録媒体から識別データが検出された場合には、検出された識別データに基づいてプログラムデータを記録媒体から読み出す読み出しステップと、読み出されたプログラムデータに基づいて記録媒体からデータ書換装置内部に記憶されているデータを更新するための更新データを読み出して読み出された更新データでデータ書換装置内部に記憶されているデータを更新する更新ステップとを備えたことを特徴とする制御方法を提供することを目的としている。

【0015】更に本願発明は、装着されるデータ書換装置内部に記憶されているデータを更新するための更新データが記録される更新データ記録領域と、データ書換装置に更新データを転送するためのプログラムデータが記録されるプログラムデータ記録領域と、更新データ及びプログラムデータが記録されていることを識別する識別データが記録される識別データ記録領域とを備えた記録媒体を提供することを目的としている。

【0016】このような本願発明装置によれば以下のような作用をする。予め制御手段を書き換え可能状態に制御して、書き換えを可能にするための機能は、記録媒体に記憶した書き換えプログラムの中に設けるようにした。この書き換えプログラムが制御手段の中に読み込まれて、初めて制御手段の書き換え禁止状態を解除して書き換え可能状態に制御され、この書き換えプログラムにより、新たなプログラムデータを制御手段に書き込むことにより、プログラムデータの書き換えができる。

【0017】即ち本発明では、不用意によるプログラムデータの書き換えまたは消去を防止するための安全性を考慮すると共に、携帯情報端末の制御手段としてのマイクロコンピュータの動作を安定な状態にする。

【0018】

【発明の実施の形態】以下、本発明の実施の形態について説明する。なお、実施の形態としての記憶媒体は板状の外形形状を有する板状メモリとし、また本発明の携帯

情報端末装置は、板状メモリに対してファイルの記録再生を行うことのできる携帯端末が相当するものとする。

【0019】〔板状メモリの外形形状〕まず、本例の記憶媒体である、板状メモリの外形形状について説明する。板状メモリは、例えば板状の筐体内部に例えば所定容量のメモリ素子を備える。本例としては、このメモリ素子としてフラッシュメモリ（Flash Memory）が用いられるものである。

【0020】筐体の正面下部から底面側にかけて例えば9個の電極を持つ端子部が形成されており、この端子部から、後述する携帯端末本体側の着脱機構を介して内部のメモリ素子に対する読出又は書込動作が行われる。筐体の平面方向の左上部は切欠部とされる。この切欠部は、この板状メモリを、例えば携帯端末本体側の着脱機構へ装填する際に挿入方向を誤ることを防止するためのものとなる。さらに底面側には、記憶内容の誤消去を防止する目的のスライドスイッチが形成されている。

【0021】〔板状メモリのフォーマット〕

〔メモリファイルシステム処理階層〕続いて、板状メモリを記憶媒体とするシステムにおけるフォーマットについて説明していく。図1は、板状メモリを記憶媒体とするシステムのファイルシステム処理階層を示すものである。図1に示すように、ファイルシステム処理階層としては、アプリケーション処理層60の下に、順次、ファイル管理処理層61、論理アドレス層62、物理アドレス層63、フラッシュメモリアクセス層64がおかれる。この階層では、ファイル管理処理層61がいわゆるFAT（File Allocation Table）となる。また、この図から分かるように、本例のファイルシステムでは論理アドレス及び物理アドレスという概念が導入されているが、これについては後述する。

【0022】特に、本実施の形態では、アプリケーション処理層60において、専用機器により後述するプログラムデータの書き換えを実行する書き換えプログラムを記憶したプログラムローダーファイルおよびプログラムデータを記憶したROMデータファイルのアドレス生成の処理の入力操作を行うことによりアドレスファイルの処理を開始するアドレスファイル処理部65が設けられ、この場合には、直ちにたとえばRAM内に置かれた物理アドレステーブル67を参照して所定のインデックスファイルの論理アドレスに対応するフラッシュメモリ上の物理アドレスを取得して、フラッシュメモリアクセス層64において設けられる物理アドレス書き込み処理部66により物理アドレスの書き込みが行われる。その後、専用機器から取り出された板状メモリに対して後述する情報端末装置においてプログラムローダーファイルを用いてROMデータファイルの書き込み処理が行われる。

【0023】このように、アドレスファイル処理においては、ファイル管理処理層61、論理アドレス層62お

よび物理アドレス層63を使用せずに、直ちにフラッシュメモリアクセス層64を行うことができるので、FATファイルシステムにおけるファイルを使用するにもかかわらず、FATファイルシステムの複雑な処理を必要とせず、処理時間を短くすることができる。なお、上述した専用機器とは、例えばFATファイルシステム等の論理アドレスと物理アドレスとのアドレス管理を行うファイルシステムを用いてRAMで構成される物理アドレステーブル67を参照して板状メモリに物理アドレスの書き込みを行う機能を有するものをいう。

【0024】本実施の形態では、特に、専用機器において、物理アドレステーブル67を設け、操作部のアドレスファイルの処理を開始するキーが押下されることによりマイクロコンピュータの機能として実現されるアドレスファイル処理を開始するアドレスファイル処理部65により、直ちに物理アドレステーブル67を参照して所定のプログラムローダーファイルおよびROMデータファイルの論理アドレスに対応するフラッシュメモリ上の物理アドレスを取得して、マイクロコンピュータの機能として実現される物理アドレス書き込み処理部66によりフラッシュメモリが内蔵された板状メモリ7に対してアクセスを行うことによりプログラムローダーファイルおよびROMデータファイルの物理アドレスの書き込みが行われる。

【0025】なお、記録データファイルまたは再生データファイルの記録または再生処理については、アプリケーション処理層60、ファイル管理処理層61、論理アドレス層62、物理アドレス層63、フラッシュメモリアクセス層64がすべて使用される。

【0026】〔物理的データ構造〕板状メモリ内の記憶素子である、フラッシュメモリの物理的データ構造を説明する。フラッシュメモリとしての記憶領域は、セグメントという固定長のデータ単位が大元となる。このセグメントは、1セグメントあたり4MB或いは8MBとして規定されるサイズであり、1つのフラッシュメモリ内におけるセグメント数は、そのフラッシュメモリの容量に依存して異なってくる。

【0027】そして、この1セグメントを、ブロックという固定長のデータ単位として8KB又は16KBにより区切るようにされる。原則として、1セグメントは512ブロックに区切られることから、ブロックnについては、 $n=511$ とされることになる。但し、フラッシュメモリでは、書き込み不可な損傷エリアであるディフェクトエリアとしてのブロック数が所定数の範囲で許可されているため、データ書き込みが有効とされる実質的なブロック数を対象とすれば、上記nは511よりも少なくなる。

【0028】形成されるブロック0～nのうち、先頭の2つのブロック0、1はブートブロックといわれる。但し、実際には有効なブロックの先頭から2つのブロック

がブートブロックとして規定されることになっており、必ずしもブートブロックがブロック0、1である保証はない。そして、残りのブロックが、ユーザデータが格納されるユーザブロックとなる。

【0029】1ブロックは、ページ0～mにより分割される。1ページの容量は、512バイトのデータエリアと16バイトの冗長部よりなる、528(=512+16)バイトの固定長とされる。なお、冗長部の構造については後述する。また、1ブロック内のページ数としては、1ブロックの容量が8KBの場合には16ページ、16KBの場合には32ページとなる。

【0030】このような、ブロック内のページ構造は、上記ブートブロックとユーザブロックとで共通である。また、フラッシュメモリでは、データの読み出しおよび書き込みはページ単位で行われ、データの消去はブロック単位で行われるものとされる。そして、データの書き込みは、消去済みのページに対してしか行われないものとされている。従って、実際のデータの書き換えや書き込みは、ブロック単位を対象として行われることになる。

【0031】先頭のブートブロックは、ページ0に対してヘッダーが格納され、ページ1には初期不良データの位置としてアドレスを示す情報が格納される。また、ページ2にはCIS/IDI(Card Information Structure/Identify Drive Information)といわれる情報が格納される。2つめのブートブロックは、ブートブロックとしてのバックアップのための領域とされている。

【0032】〔物理アドレス及び論理アドレスの概念〕次に、上述したようなフラッシュメモリの物理的データ構造を踏まえたうえで、データ書き換え動作に従って、本例のファイルシステムにおける物理アドレスと論理アドレスの概念について説明する。

【0033】各ブロックに対しては物理アドレスが付される。この物理アドレスはメモリにおけるブロックの物理的な配列順に従って決まるもので、或るブロックとこれに対応付けされた物理アドレスとの関係は不変となる。

【0034】ここで、ブロックデータの記憶されている使用ブロックと、ブロックデータが消去された即ち、未記録領域である未使用ブロックとがある。

【0035】そして、論理アドレスは、ブロックに対して書き込まれたデータに付随するようにして割り振られるアドレスとされる。そして、この論理アドレスが、後述するFATファイルシステムが利用するアドレスとされている。

【0036】ここで、上記状態から、物理アドレスに格納されているデータの更新として、内容の書き換え又は一部消去を行うとする。このような場合、フラッシュメモリのファイルシステムでは、同じブロックに対して更

新したデータを再度書き込むことはせずに、未使用のブロックに対してその更新したデータを書き込むようにされる。

【0037】そして、データ更新前の状態では更新前の物理アドレスに対応していた論理アドレスが、更新されたデータが書き込まれたブロックの更新後の物理アドレスに対応するように、論理アドレスについての変更を行うものである。

【0038】つまり、物理アドレスはブロックに対して固有に付されるアドレスであり、論理アドレスは、一旦ブロックに対して書き込まれたデータに付随するようにしてついて回る、ブロック単位の書き込みデータに固有となるアドレスであるといえることができる。

【0039】このようなブロックのスワップ処理が行われることで、或る同一の記憶領域であるブロックに対して繰り返し集中的にアクセスされることがなくなり、書き換え回数の上限があるフラッシュメモリの寿命を延ばすことが可能となる。

【0040】そして、この際に論理アドレスを上記のようにして扱うことで、ブロックのスワップ処理によって更新前と更新後のデータとで書き込まれるブロックの移動があるようにされても、FATからは同一のアドレスが見えることになり、以降のアクセスを適正に実行することができるものである。

【0041】なお、後述する論理-物理アドレス変換テーブル上での更新のための管理を簡略にすることなどを目的として、ブロックのスワップ処理は、1セグメント内で完結するものとして規定されている。逆に言えば、ブロックのスワップ処理はセグメント間で跨るようには行われない。

【0042】〔論理-物理アドレス変換テーブル〕上記説明から分かるように、ブロックのスワップ処理が行われることで、物理アドレスと論理アドレスの対応は変化する。従って、フラッシュメモリに対するデータの書き込み及び読み出しのためのアクセスを実現するには、物理アドレスと論理アドレスとの対応が示される論理-物理アドレス変換テーブルが必要となる。つまり、論理-物理アドレス変換テーブルをFATが参照することで、FATが指定した論理アドレスに対応する物理アドレスが特定され、この特定された物理アドレスにより示されるブロックにアクセスすることが可能になるものである。逆に言えば、論理-物理アドレス変換テーブルが無ければ、FATによるフラッシュメモリへのアクセスが不可能となる。

【0043】論理-物理アドレス変換テーブルの構造例を示す。論理-物理アドレス変換テーブルは、フラッシュメモリの最後のセグメント内であることが規定される或るブロックに対して、格納される。まず、ブロックを分割するページのうち、ページ0、1からなる2ページの領域がセグメント0用の論理-物理アドレス変換テ

ブルとして割り当てられる。例えば、フラッシュメモリが4MBの容量であれば1セグメントしか有さないために、このページ0、1のみの領域が論理-物理アドレス変換テーブルの領域となる。

【0044】また、例えばフラッシュメモリが8MBの容量であれば2セグメントを有するため、セグメント0用の論理-物理アドレス変換テーブルとして割り当てられるページ0、1に加え、これに続くページ2、3の2ページがセグメント1用の論理-物理アドレス変換テーブルとして割り当てられることになる。

【0045】以降、フラッシュメモリの容量の増加に応じて、続く2ページごとにセグメントごとの論理-物理アドレス変換テーブルの割り当て領域が設定されていくことになる。そして、最大の128MBの容量を有する場合であれば16セグメントが存在するため、最大では、セグメント15用までのページが論理-物理アドレス変換テーブルの領域として割り当てられることになる。従って、最大の128MBの容量のフラッシュメモリでは、30ページが使用されることになり、ページNとしては、最大でN=29となる。これまでの説明から分かるように、論理-物理アドレス変換テーブルは、セグメントごとに管理されるものである。

【0046】1セグメントあたりの論理-物理アドレス変換テーブルの構造を示すものとして、2ページ分のデータエリアを抜き出して示している。つまり、1ページのデータエリアは512バイトであることから、1024(=512×2)バイトが展開されている。

【0047】ここで、フラッシュメモリ容量と論理-物理アドレス変換テーブルのサイズとの関係を説明しておく。説明したように、1セグメントを管理するための論理-物理アドレス変換テーブルのサイズは2ページ分の1024バイト、つまり1KBとなる。従って、フラッシュメモリが1セグメントにあたる4MBの容量では論理-物理アドレス変換テーブルは1KBのサイズとなる。また、フラッシュメモリの容量が2セグメントにあたる8MBでは論理-物理アドレス変換テーブルは4ページにあたる2KBとなる。

【0048】また、フラッシュメモリの容量が16MBの場合、2048ブロック=4セグメントのものでは論理-物理アドレス変換テーブルは8ページにあたる4KB、1024ブロック=2セグメントのものでは論理-物理アドレス変換テーブルは4ページにあたる2KBとなる。

【0049】そして、フラッシュメモリの容量が4セグメントにあたる32MBでは論理-物理アドレス変換テーブルは8ページにあたる4KB、フラッシュメモリの容量が8セグメントにあたる64MBでは論理-物理アドレス変換テーブルは16ページにあたる8KBとなり、フラッシュメモリの容量が最大の16セグメントにあたる128MBでは論理-物理アドレス変換テーブル

は32ページにあたる16KBとなる。

【0050】〔携帯情報端末および板状メモリの構成〕
続いて図2で本例の携帯情報端末および板状メモリの構成を説明する。図2は、これまで説明した板状メモリに対応してデータの読出・書込・編集を行うことの出来る携帯情報端末のセット本体である本体装置の構成を示している。この図に示すセット本体20と板状メモリ7とにより、ファイル記憶システムが構成される。

【0051】なお、セット本体20が、板状メモリ7に対する書込や読出の対象として扱うことのできる主データの種別は多様であり、例えば動画データ、静止画データ、ボイスデータ等のオーディオデータ、CD (Compact Disc) やMD (Mini Disc TM) 等から再生されるような高音質なオーディオデータ、制御用データなどがある。

【0052】本例では、説明の簡略化のため、主データとしてのスピーチデータを記憶および再生するシステムとして説明していくが、セット本体20内に、動画、静止画、オーディオ等のデータの入出力系および処理系を備えることにより、これらの動画等のデータファイルの記憶システムとできることはいうまでもない。

【0053】セット本体20の構成としては、当該セット本体20が着脱可能に装填される着脱機構13が備えられ、この着脱機構13に装填された板状メモリ7と、制御用の組み込み型マイクロコンピュータ1とのデータの授受は、図示しないホストインターフェイスICを介することで行われる。

【0054】また、このセット本体20には、例えば、図示しないマイクロフォンが備えられて、このマイクロフォンにより収音されたスピーチやディクテーション等のオーディオ信号は、マイクアンプを介してオーディオ信号としてDSP (Digital Signal Processor) に対して供給される。DSPでは、この入力されたオーディオ信号をデジタルオーディオデータに変換して、エンコード処理等をはじめとする所要の信号処理を施し、記録データとしてマイクロコンピュータ1に供給する。マイクロコンピュータ1は、この記録データをホストインターフェイスICを介して板状メモリ7に記録するための処理を実行することが可能とされる。

【0055】また、マイクロコンピュータ1は、板状メモリ7に記録されているオーディオデータをホストインターフェイスICを介して読み出し、この読み出したデータをDSPに供給する。DSPでは、供給されたデータについて復調処理をはじめとする所要の信号処理を施して、最終的にはアナログオーディオ信号としてスピーカアンプに出力する。スピーカアンプでは、入力されたオーディオ信号を増幅してスピーカに出力する。これにより、再生オーディオが出力される。

【0056】また、マイクロコンピュータ1は、表示

ライバを制御することで、表示部に対して、所要の画像を表示させることが可能とされる。例えばユーザーの操作のためのメニューやガイド表示、或いは板状メモリ7に記憶されたファイル内容などの表示が実行される。また、例えば板状メモリ7に対して動画若しくは静止画の画像データが記録されているとすれば、この画像データを読み出して、表示部に表示させるようにすることも可能とされる。

【0057】ここで、本実施の形態では、マイクロコンピュータは、制御用のプログラムデータを記憶したROMデータファイルを記憶するフラッシュROM2と、制御データおよびプログラムデータの書き換えを実行する書き換えプログラムを記憶したプログラムローダーファイルを記憶する内部RAM3と、フラッシュROM2に記憶されたプログラムデータの書き換え機能を禁止状態または有効状態に制御可能なライトイネーブル端子4と、出力端子5と、ライトイネーブル端子4と出力端子5とを接続してプログラムデータの書き換え機能を禁止状態あるいは許可状態の制御信号の伝達に用いられるジャンパー線14とを有して構成される。また、マイクロコンピュータ1には、プログラムデータの書き換えを実行する書き換えプログラムを記憶したプログラムローダーファイルを一時的に記憶する外部RAM6が外部に接続されている。

【0058】このように、フラッシュROM2の書き換え機能を有するマイクロコンピュータ1はフラッシュROM2の書き換えの可否をハードウェアの機能として制御するライトイネーブル端子4を備えている。そしてこのライトイネーブル端子4をコントロールするためにマイクロコンピュータの出力端子5がジャンパー線14により接続されていて、通常はライトイネーブル端子4を書き換え禁止状態に制御している。ここで、出力端子5からジャンパー線14を介してライトイネーブル端子4を書き換え可能状態に制御して、フラッシュROM2の書き換えを可能にするための機能は、初期状態ではセット本体20側のフラッシュROM2のROMデータには存在していない。

【0059】本実施の形態では、不用意によるプログラムデータの書き換えまたは消去を防止するための安全性を考慮して、出力端子5からあらかじめ例えばプリント基板のプリントパターンによって配線されたジャンパー線14を介してライトイネーブル端子4を書き換え可能状態に制御して、フラッシュROM2の書き換えを可能にするための機能は、板状メモリ7に記憶したソフトウェアモジュールで構成されるプログラムローダーの中に設けるようにした。このプログラムローダーがマイクロコンピュータ1の内部RAM3の中に読み込まれて、初めてライトイネーブル端子4の書き換え禁止状態を解除して書き換え可能状態に制御され、このプログラムローダーにより、新たなROMデータファイルをフラッシュ

ROM2に書き込むことにより、ROMデータの書き換えができるように構成されている。

【0060】また、本実施の形態では、板状メモリ7側には、プログラムデータの書き換えを実行する書き換えプログラムを記憶したプログラムローダーファイル9と、新たな制御用のプログラムデータを記憶したROMデータファイル11とが記憶されている。

【0061】また、本実施の形態では、いま携帯情報端末本体20に脱着機構13を介して接続されている板状メモリ7には、この板状メモリが正規のプログラムデータ書き換え用のプログラムであるプログラムローダーファイル9およびROMデータファイル11を記憶しているか否かを識別するローダー用板状メモリ識別ファイル8が設けられている。

【0062】このように、板状メモリ7にローダー用板状メモリ識別ファイル8が設けられていること、およびセット本体20側ではなく板状メモリ7側にプログラムローダーファイル9が設けられることにより、携帯情報端末本体20の電源が不安定となることに起因するマイクロコンピュータ1の暴走等の予期しない動作により、ROMデータが書き換えられて破壊されることを防ぐことができる。

【0063】また、本実施の形態では、板状メモリ7にはプログラムローダーファイル9のアドレスを記憶したプログラムローダーアドレスファイル10と、ROMデータファイル11のアドレスを記憶したROMデータアドレスファイル12とが設けられている。

【0064】ここで、特に、プログラムローダーアドレスファイル10およびROMデータアドレスファイル12の物理アドレスの書き込みは以下のようにして行われる。図1において説明したように、専用機器において、たとえばRAM上に物理アドレス変換テーブル67を設け、操作部のアドレスファイルの処理を開始するキーが押下されることによりアドレスファイル処理を開始するマイクロコンピュータの機能として実現されるアドレスファイル処理部65により、直ちに物理アドレステーブル67を参照して所定のプログラムローダーファイルおよびROMデータファイルの論理アドレスに対応するフラッシュメモリ上の物理アドレスを取得して、マイクロコンピュータの機能として実現される物理アドレス書き込み処理部66により板状メモリ7に対してアクセスを行うことによりプログラムローダーファイルおよびROMデータファイルの物理アドレスの書き込みを行う。

【0065】なお、図示しない操作部には、セット本体20に対する各種操作をユーザが行うための各種キーが設けられている。マイクロコンピュータ1は、この操作部に対して行われた操作に応じたコマンドを受信し、コマンドに応じた所要の制御処理を実行する。操作内容としては、ROMデータ書き換え指示、ファイル記憶指示、ファイル選択指示、ファイル再生指示、編集指示な

どが可能とされる。

【0066】なお、図2に示すセット本体20の構成はあくまでも一例であり、これに限定されるものではない。つまり、板状メモリ7に対応してデータの送受信が可能な構成を採る限りは、どのようなタイプの電子機器とされていても構わないものである。

【0067】次に、図3を使い動作の概要を説明する。なお、図3に示した各部のうち図2と同一のブロックについては同一の符号を付している。マイクロコンピュータ1にはフラッシュROM2とRAM3とが備えられており、図3においては機能ブロックとして説明するためにフラッシュROM2をフラッシュROMAとフラッシュROMBとに分離して描かれている。しかしこれは物理的に分離している必要はなく同一のフラッシュROMを例えばブロックアドレスで分けて使用するなどして実現することが可能である。

【0068】板状メモリ7には図2に示されているブロックと同様に識別ファイル8、プログラムローダーアドレスファイル10、プログラムローダーファイル9、ROMデータアドレスファイル12、ROMデータファイル11とがそれぞれ記憶されている。

【0069】初期状態としてはマイクロコンピュータ1のフラッシュROMには機能の追加前の制御プログラムが記憶されており、RAM3には機能追加前の制御プログラムが使用したデータ等が記憶されている。

【0070】マイクロコンピュータ1のフラッシュROMAにはフラッシュROMBに記憶されている機能追加前の制御プログラムを書き換えるための板状メモリ7に記憶されているプログラムが記憶されているか否かの判別と、書き換えるためのプログラムとしてのプログラムローダーアドレスファイル並びにプログラムローダーをロードしてくるためのプログラムとして書換プログラムローダー15が記憶されている。

【0071】携帯情報端末20に識別ファイル8、プログラムローダーアドレスファイル10、プログラムローダーファイル9、ROMデータアドレスファイル12、ROMデータファイル11とがそれぞれ記憶されている板状メモリ7が装着されるとともに、プログラムのアップデートの指示が入力されるとフラッシュROMAに記憶された書換プログラムローダー15が実行される。書換プログラムローダー15は、(1)のように板状メモリ7を物理アドレスでアクセスしながら識別ファイル8をサーチする。識別ファイル8が検出されると現在携帯情報端末20に装着されている板状メモリ7には、携帯情報端末20のマイクロコンピュータ1内部に設けられたフラッシュROMに記憶されている制御プログラムをアップデートするためのデータ及びデータの書換プログラムが記憶されていると判断される。

【0072】識別ファイル8が検出されると書換プログラムローダー15は、(2)のように板状メモリ7の内

部を物理アドレスでアクセスしてプログラムローダーアドレスファイル10をサーチする。プログラムローダーアドレスファイル10が検出されると書換プログラムローダー15は、(3)のようにプログラムローダーアドレスファイル10に基づいて板状メモリ7に記憶されたプログラムローダー9の記憶されている位置を特定し、(4)のように特定されたアドレスからプログラムローダー9をマイクロコンピュータ1の内部に設けられているRAM3にロードする。

【0073】すべてのプログラムローダー9がRAM3にロードされた後、書換プログラムローダー15はRAM3上に展開されたプログラムローダー16に制御を渡す。具体的にはRAM3上に展開されたプログラムローダー16の実行開始番地からプログラムがマイクロコンピュータ1によって実行されるようにプログラムローダー16の実行開始番地にジャンプする命令をマイクロコンピュータ1に実行させることになる。

【0074】実行制御権を得たRAM3上のプログラムローダー16は、(5)のように板状メモリ7を物理アドレスでアクセスしてROMアドレスファイルをサーチする。RAM3上のプログラムローダー16は、(6)のようにROMデータアドレスファイル12が検出されるとROMデータアドレスファイル12に従って(7)のように板状メモリ7からROMデータファイル11をアクセスして読み込まれる書き換え用データ14を、マイクロコンピュータ1のフラッシュROMBのアップデートされるプログラムが記録された領域を消去しながらフラッシュROMBに書き込んでいく。

【0075】板状メモリ7に記憶されたROMデータファイル11のうちの必要なデータがマイクロコンピュータ1のフラッシュROMBにすべて書き込まれるとRAM3に記憶されているプログラムローダー16は、マイクロコンピュータがイニシャライズ動作をするように例えばRESET命令を実行してフラッシュROM2に記録された携帯情報端末20を制御するプログラムが動作するようにマイクロコンピュータ1を制御する。

【0076】マイクロコンピュータ1はイニシャライズプログラムを実行することによりRAM3上に記憶されていたプログラムローダー16を消去して通常の処理動作に戻るようになる。

【0077】[フラッシュROM書き換え動作のフローチャート] 次に、図4のフラッシュROM書き換え動作のフローチャートを用いて本実施の形態の動作を説明する。図4においては、図2に示したROMデータの書き換え禁止状態に制御された携帯情報端末20に対して板状メモリ7に記憶されたプログラムローダーファイル9およびROMデータファイル11を用いてROMデータの書き換えを行う場合について説明する。

【0078】図4において、フラッシュROM書き換え処理を開始して、ステップS1でマイクロコンピュータ

1は携帯情報端末本体に接続された板状メモリ7にローダー用板状メモリ識別ファイル8が記憶されているか否かを判断する。具体的には、マイクロコンピュータ1は、ローダー用板状メモリ識別ファイル8に、この板状メモリが正規のプログラムデータ書き換え用のプログラムローダーファイル9およびROMデータファイル11を記憶していることを識別するための識別子としてのヘッダーが設けられているか否かを検出する。

【0079】図6に、物理アドレスファイル構成を示す。図6において、上述した物理アドレスによるフラッシュメモリへの書き込み、読み出しまたは消去のためのアクセスは、以下のように実行される。まず、第1に、書き込み、読み出しまたは消去のアクセスを行う正しいヘッダー40が来るまで物理アドレス順にブロックの先頭から検索を行う。第2に、第1の処理で正しいヘッダー40が検索されたら、これに対応する物理アドレス41を取得する。このようにして所望の物理アドレスに対してアクセスが実行される。

【0080】図7A、図7B、図7Cに、識別ファイルのヘッダーデータの例を示す。プログラムローダーファイル9およびROMデータファイル11のロード方法として、携帯情報端末本体にインターフェースを介してパーソナルコンピュータが接続されてパーソナルコンピュータからロードする場合と、携帯情報端末本体に板状メモリが接続されて板状メモリからロードする場合とがある。

【0081】図7Aにおいて、パーソナルコンピュータからのロードの場合には、識別データ50として、ABC DEF MEMSTICK PROGRAM LOADER LOAD FROM PCが記憶され、スペース51に続いて1ブロックあたりの記憶領域を示すページ数52として32が記録される。

【0082】図7Bにおいて、板状メモリからのロードの場合には、識別データ53として、ABC DEF MEMSTICK PROGRAM LOADER LOAD FROM MSが記憶され、スペース54に続いて1ブロックあたりの記憶領域を示すページ数55として16が記録される。

【0083】このような識別子としてのヘッダーデータを検出して正規のROMデータが記録された板状メモリであることがマイクロコンピュータ1により認識されると、ステップS2へ進む。

【0084】次に、ステップS2で、マイクロコンピュータ1は、ソフトウェアモジュールとしてプログラムローダーのアドレスファイルであるプログラムローダーアドレスファイル10を読み込む。具体的には、図1において説明したように、専用機器において、RAM上に物理アドレステーブル67を設け、操作部のアドレスファイルの処理を開始するキーが押下されることによりアドレスファイル処理を開始するマイクロコンピュータ1の

機能として実現されるアドレスファイル処理部65により、直ちに物理アドレステーブル67を参照して所定のプログラムローダーファイルの論理アドレスに対応するフラッシュメモリ上の物理アドレスを取得して、マイクロコンピュータ1の機能として実現される物理アドレス書き込み処理部66により板状メモリ7に対してアクセスを行うことによりプログラムローダーファイルの物理アドレスの書き込みを行う。このプログラムローダーアドレスファイル10を読み込むことにより、携帯情報端末20のマイクロコンピュータ1は、FAT等の大規模なプログラムを解釈する必要が無くなる。

【0085】ステップS3で、マイクロコンピュータ1は、ソフトウェアモジュールとしてプログラムローダーが記憶されたプログラムローダーファイル9を内部RAM3上に読み込む。具体的には、マイクロコンピュータ1は、プログラムローダーアドレスファイル10の物理アドレスデータを用いて、板状メモリ7の記憶領域をアクセスして、ソフトウェアモジュールとしてプログラムローダーが記憶されたプログラムローダーファイル9を読み出して、内部RAM3上の所定領域に書き込む。

【0086】図7Cに、プログラムローダーファイルのヘッダーデータの例を示す。図7Cにおいて、識別データ56として、ABC DEF MEMSTICK PROGRAM LOADERが記憶され、ファイル名57としてromdata.datが記録され、スペース58に続いて物理ブロックアドレスデータ59として0×00A3、0×003F、0×FFFFが記録される。

【0087】ステップS4で、マイクロコンピュータ1は、ROMデータファイルのアドレスファイルであるROMデータアドレスファイル12を読み込む。具体的には、図1において説明したように、専用機器において、RAM上に物理アドレス変換テーブル67を設け、操作部のアドレスファイルの処理を開始するキーが押下されることによりアドレスファイル処理を開始するマイクロコンピュータ1の機能として実現されるアドレスファイル処理部65により、直ちにRAM上の物理アドレステーブル67を参照して所定のROMデータファイルの論理アドレスに対応するフラッシュメモリ上の物理アドレスを取得して、マイクロコンピュータ1の機能として実現される物理アドレス書き込み処理部66により板状メモリ7に対してアクセスを行うことによりROMデータアドレスファイルの物理アドレスの書き込みを行う。このROMデータアドレスファイル12を読み込むことにより、携帯情報端末20の組み込みマイクロコンピュータ1は、FAT等の大規模なプログラムを解釈する必要が無くなる。

【0088】なお、上述したプログラムローダーファイル9およびプログラムローダーアドレスファイル10は、FATファイルシステム上に記録されているので、

パーソナルコンピュータから携帯情報端末20に容易に書き込むことができる。ステップS5で、マイクロコンピュータ1は、ROMデータが記憶されたROMデータファイル11を外部RAM6上に読み込む。具体的には、マイクロコンピュータ1は、ROMデータファイル11の物理アドレスデータを用いて、板状メモリ7の記憶領域をアクセスして、ROMデータが記憶されたROMデータファイル11を読み出して、外部RAM6上の所定領域に書き込む。

10 【0089】図7Cに、ROMデータファイルのヘッダーデータの例を示す。図7Cにおいて、識別データ56として、ABC DEF MEMSTICK PROGRAM LOADERが記憶され、ファイル名57としてromdata.datが記録され、スペース58に続いて物理ブロックアドレスデータ59として0×00A3、0×003F、0×FFFFが記録される。

20 【0090】ステップS6で、マイクロコンピュータ1は、内部RAM上のプログラムローダーを起動する。具体的には、マイクロコンピュータ1は、内部RAMに格納したプログラムローダーを起動して、出力端子5を制御して、出力端子5からジャンパー線14を介してライトイネーブル端子4に対して書き換え可能状態に制御するための制御信号を出力する。

【0091】ステップS7で、マイクロコンピュータ1は、ライトイネーブル端子4を書き込み可能状態に変更する。具体的には、マイクロコンピュータ1は、出力端子5から出力される制御信号に基づいてライトイネーブル端子4が書き換え可能状態になるように出力端子5を制御する。

30 【0092】このようにして、フラッシュROM2の書き換えを可能にするための機能は、板状メモリ7に記憶したソフトウェアモジュールで構成されるプログラムローダーの中に設けるようにしているので、このプログラムローダーがマイクロコンピュータ1の内部RAM3の中に読み込まれて、初めてライトイネーブル端子4の書き換え禁止状態を解除して書き換え可能状態に制御することができる。

40 【0093】ステップS8で、マイクロコンピュータ1は、外部RAM上のROMデータファイルを書き込む。具体的には、マイクロコンピュータ1は、外部RAM6に格納したROMデータをフラッシュROM2へ書き込む。

【0094】このようにして、プログラムローダーにより、新たなROMデータファイルをフラッシュROM2に書き込むことにより、ROMデータの書き換えが実行される。

50 【0095】[必要なデータ作成動作のフローチャート]次に、図5の必要なデータ作成動作のフローチャートを用いて本実施の形態の動作を説明する。図5においては、図2に示した板状メモリ7に記憶されるローダー

用板状メモリ識別用ファイル8、プログラムローダーファイル9、ROMデータファイル11、プログラムローダーアドレスファイル10、ROMデータアドレスファイル12を専用機器またはパーソナルコンピュータなどの外部装置により作成する場合について説明する。

【0096】図5において、データ作成処理を開始して、ステップS11で板状メモリを専用機器で初期化する。具体的には、板状メモリをFAT等のファイルシステム用にフォーマット可能な専用機器に接続して不要なデータ等を消去するように初期化する。

【0097】ステップS12で、板状メモリをパーソナルコンピュータへ接続する。具体的には、板状メモリを所定のインターフェースを介してパーソナルコンピュータへ接続する。

【0098】ステップS13で、プログラムローダーファイルをパーソナルコンピュータから書き込む。具体的には、板状メモリを専用機器でFAT等のファイルシステム用にフォーマットすることにより、FATファイルシステム上に予めパーソナルコンピュータ上で作成されたプログラムローダーファイルを記録することができるので、パーソナルコンピュータから板状メモリ7に容易にプログラムローダーファイルを書き込むことができる。

【0099】図7Cに、プログラムローダーファイルのヘッダーデータの例を示す。図7Cにおいて、識別データ56として、ABC DEF MEMSTICK PROGRAM LOADERが記憶され、ファイル名57としてromdata.datが記録され、スペース58に続いて物理ブロックアドレスデータ59として0×00A3、0×003F、0×FFFFFFが記録される。

【0100】ステップS14で、ROMデータファイルをパーソナルコンピュータから書き込む。具体的には、板状メモリを専用機器でFAT等のファイルシステム用にフォーマットすることにより、FATファイルシステム上に予めパーソナルコンピュータ上で作成されたROMデータファイルを記録することができるので、パーソナルコンピュータから板状メモリ7に容易にROMデータファイルを書き込むことができる。

【0101】図7Cに、ROMデータファイルのヘッダーデータの例を示す。図7Cにおいて、識別データ56として、ABC DEF MEMSTICK PROGRAM LOADERが記憶され、ファイル名57としてromdata.datが記録され、スペース58に続いて物理ブロックアドレスデータ59として0×00A3、0×003F、0×FFFFFFが記録される。

【0102】ステップS15で、板状メモリをパーソナルコンピュータから外す。具体的には、板状メモリを所定のインターフェースから外すことにより、パーソナルコンピュータから外す。

【0103】ステップS16で、専用機器にてローダー用板状メモリ識別用ファイルを作成する。具体的には、図6に示した、物理アドレスファイル構成において、ヘッダー40の位置に、例えば、図7Aに示した、識別ファイルのヘッダーデータとして、パーソナルコンピュータからのロードの場合には、識別データ50として、ABC DEF MEMSTICK PROGRAM LOADER LOADFROM PCが記憶され、スペース51に続いて1ブロックあたりの記憶領域を示すページ数52として32が記録される。また、図7Bにおいて、板状メモリからのロードの場合には、識別データ53として、ABC DEF MEMSTICK PROGRAM LOADER LOAD FROM MSが記憶され、スペース54に続いて1ブロックあたりの記憶領域を示すページ数55として16が記録される。

【0104】ステップS17において、専用機器にてプログラムローダーファイルのアドレスファイルを作成する。具体的には、図1において説明したように、専用機器において、たとえばRAM内に物理アドレステーブル67を設け、操作部のアドレスファイルの処理を開始するキーが押下されることによりアドレスファイル処理を開始するマイクロコンピュータ1の機能によって実現されるアドレスファイル処理部65により、直ちに物理アドレステーブル67を参照して所定のプログラムローダーファイルの論理アドレスに対応するフラッシュメモリ上の物理アドレスを取得して、マイクロコンピュータ1の機能によって実現される物理アドレス書き込み処理部66により板状メモリ7に対してアクセスを行うことによりプログラムローダーファイルの物理アドレスの書き込みを行う。

【0105】ステップS18において、専用機器にてROMデータファイルのアドレスファイルを作成する。具体的には、図1において説明したように、専用機器において、たとえばRAM内に物理アドレステーブル67を設け、操作部のアドレスファイルの処理を開始するキーが押下されることによりアドレスファイル処理を開始するマイクロコンピュータ1の機能によって実現されるアドレスファイル処理部65により、直ちに物理アドレステーブル67を参照して所定のROMデータファイルの論理アドレスに対応するフラッシュメモリ上の物理アドレスを取得して、マイクロコンピュータ1の機能によって実現される物理アドレス書き込み処理部66により板状メモリ7に対してアクセスを行うことによりROMデータファイルの物理アドレスの書き込みを行う。

【0106】次に図8を用いて図2においてマイクロコンピュータ1に内蔵されたフラッシュROM2に記憶されたプログラムを書き換える動作を説明する。図8は板状メモリ7内部のROMに記憶されたデータを物理ブロックアドレスを付して示している。

【0107】図8の物理ブロックアドレスの0×000

4には図7Bの識別用ファイルが記録されている。これは、図2の板状メモリ7内に記憶されているローダー用板状メモリ識別ファイルである。

【0108】図8の物理ブロックアドレスの0×0052には図7CのROMデータアドレスファイルが記録されている。これは、図2の板状メモリ7内に記憶されているROMデータアドレスファイル12である。

【0109】図8の物理ブロックアドレスの0×00A2には図7Dのプログラムローダーアドレスファイルが記録されている。これは、図2の板状メモリ7内に記憶されているプログラムローダーアドレスファイル10である。

【0110】また、図8の物理ブロックアドレスの0×0041は図2の板状メモリ7内に記憶されているプログラムローダーファイル9である。

【0111】図8の物理ブロックアドレスの0×00A3と0×003Fに記録されているデータは、図2の板状メモリ7内に記憶されているROMデータファイル11を構成するデータである。

【0112】携帯情報端末20に設けられたマイクロコンピュータ1に内蔵されたフラッシュROM2に記憶されたプログラムを書き換える場合、最初にフラッシュROM2に記憶されているプログラム書換の準備を行うプログラムが起動される。

【0113】このプログラム書換の準備を行うプログラムは図8に示した板状メモリ7内を物理ブロックアドレスで最初のアドレスから順に指示してアクセスしていき、物理ブロックアドレスで指示したブロックから読み出されるデータの所定位置に図7Bに示したようなローダー用板状メモリ識別ファイル8であることを示すヘッダーが有るか否かを判別していく。図8のようにデータが記録されている場合には、物理ブロックアドレスの0×0004を読み出したときに、装着された板状メモリに携帯情報端末20に設けられたマイクロコンピュータ1に内蔵されたフラッシュROM2に記憶されたプログラムを書き換えるためのプログラムとデータとが記録されていることが判別できる。つまり、物理ブロックアドレスの0×0004のブロックを読み出したときに、ABC DEF MEMSTICK PROGRAM LOADER LOAD FROM MS の文字列が確認

【0114】ローダー用板状メモリ識別ファイル8が装着されている板状メモリ7に記録されていると判別されると、続いてプログラムローダーアドレスファイル10とROMデータアドレスファイル12を板状メモリ内から検出する。

【0115】具体的には、物理ブロックアドレス0×0000から順にブロックを読み出していきプログラムロ

ーダーアドレスファイル10であれば、ABC DEF MEMSTICK PROGRAM LOADER loader. dat の文字列が記録されている図7Dに相当するブロックを検索する。また、ROMデータアドレスファイル12であれば、ABC DEF MEMSTICK PROGRAM LOADER romdata. dat の文字列が記録されている図7Cに相当するブロックを検索する。

【0116】図8のようにデータが記録されている場合には、物理ブロックアドレスの0×0052においてプログラムローダーアドレスファイルが検出され、物理ブロックアドレスの0×00A2においてROMデータアドレスファイルが検出される。

【0117】プログラムローダーアドレスファイル10が検出されるとマイクロコンピュータ1は、プログラムローダーアドレスファイル10から板状メモリ7に記録されているフラッシュROM2を書き換えるプログラムが記録されている物理アドレスを読み出す。図8の場合であれば図7Dの物理ブロックアドレスデータ7.3の位置から0×0041がまず最初に読み出される。

【0118】マイクロコンピュータ1は0×0041なる物理ブロックアドレスに基づいて板状メモリ7の0×0041のブロックからプログラムローダーファイルを読み出して内部RAM3に記憶させる。プログラムローダーアドレスファイル10から0×0041のブロックに続くプログラムローダーファイルが板状メモリ7に記録されているか否かを判別するために図7Dの物理ブロックアドレスデータ7.3の位置から次の物理ブロックアドレスデータを読み出す。この例の場合次に読み出されるデータは0×FFFFであるため物理ブロックアドレス0×0041に続くプログラムローダーファイルが無いことが判別できる。

【0119】次にマイクロコンピュータ1は検出されたROMデータアドレスファイルから図7Cの物理ブロックアドレス5.9を読み出して最初の物理ブロックアドレスとして0×00A3が読み出される。読み出された物理ブロックアドレス0×00A3に基づいてマイクロコンピュータ1は、板状メモリ7の物理ブロックアドレス0×00A3のブロックから1st ROM Dataを読み出して携帯情報端末20に設けられた外部RAM6に記憶させる。続いてマイクロコンピュータ1は図7CのROMデータアドレスファイルの物理ブロックアドレスデータ5.9から2番目の物理ブロックアドレスとして0×003Fを読み出す。読み出された物理ブロックアドレスの0×003Fに基づいてマイクロコンピュータ1は板状メモリ7の物理ブロックアドレス0×003Fから2nd ROM Dataの1ブロックを読み出して外部RAM6に記憶させる。更にマイクロコンピュータ1は図7CのROMデータアドレスファイルの物理ブロックアドレスデータ5.9から次の物理ブロックアド

レスとして0×FFFFである終端データを読み出すことでROMデータがこれ以上無いことを知りROMデータの読み出し処理を終了する。

【0120】プログラムローダーファイル9とROMデータファイル11のそれぞれのファイルが内部RAM3と外部RAM6のそれぞれに記憶されると、フラッシュROM2に記憶されている書換の準備を行うプログラムは、内部RAM3に記憶されたプログラムローダーファイルが動作するように制御権を内部RAM3に記憶されたプログラムローダーファイルに渡す。具体的には内部RAM3に記憶されたプログラムローダーの実行開始位置にジャンプするようにマイクロコンピュータ1に指示されることで行われる。

【0121】実行の制御権を得た内部RAM3に記憶されたプログラムローダーは、マイクロコンピュータ1の出力端子5を制御可能なようにプログラムされていて、ジャンパー線14を通してライトイネーブル端子4がフラッシュROM2の書換が可能な状態になるように出力端子5を制御する。

【0122】続いて内部RAM3に記憶されたプログラムローダーは、少なくとも外部RAM6に記憶されたフラッシュROMに記憶されているデータを置き換えるデータが記憶される領域の消去を行い、外部RAM6に記憶されているデータをフラッシュROM2の所定ブロックに書き込む。書き込みが終了した後は出力端子5を制御してライトイネーブル端子4をフラッシュROM2の書き換えが不可能な状態に設定し、その後ブートローダーは、フラッシュROM2に記憶されているプログラムに制御権を渡し、フラッシュROM2の書換処理を終了する。具体的にはフラッシュROM2に記憶されているプログラムのうちの初期状態に実行が開始されるアドレスにジャンプすることで行われる。

【0123】図1のファイルシステム処理階層を用いて、上述した図5のフローチャートにおけるステップS17およびステップS18のプログラムローダーファイルおよびROMデータファイルの物理アドレス取得手順を説明する。

【0124】図1において、まず、第1に、アドレスファイル処理部65において、アプリケーション処理層60のディレクトリから先頭のクラスタ番号およびFATから連続するクラスタ番号を取得する。第2に、第1の処理で得たクラスタ番号を、物理アドレステーブル67を参照することにより、物理アドレスへ変換する。これにより、所望のプログラムローダーファイルおよびROMデータファイルの物理アドレスを取得することができる。このようにして、論理アドレスのクラスタ番号から物理アドレスを算出している。

【0125】【FAT構造】図1のファイルシステム階層で説明したように、ファイル管理処理はFATにより行われることになる。即ち図2に示した構成の携帯情報

端末20により、板状メモリ7に対するデータ書込および読出である記録再生を実現するには、アプリケーション処理での要求に伴ってFATによるファイル記憶位置管理が参照され、さらに上述した論理-物理アドレス変換が行われて実際のアクセスが行われることになる。ここで、FATの構造について説明しておく。

【0126】FATによる管理構造の概要を示す。なお、本例ではFAT及び論理-物理アドレス変換テーブルは板状メモリ15内に格納されることになるが、FAT構造が、板状メモリ内での管理構造となるものである。

【0127】そして、FAT管理構造は、パーティションテーブル、空き領域、ブートセクタ、FAT、FATのコピー、ルートディレクトリ、データ領域から成る。データ領域には、クラスタ2、クラスタ3・・・として単位データを示しているが、このクラスタとは、管理単位であってFATで扱う1データ単位となる。一般にFATでは、クラスタサイズは標準で4Kバイトとされるが、このクラスタサイズは512バイト〜32Kバイトの間で2のべき乗の大きさをとることができる。

【0128】本例の板状メモリでは、上述したように1つのブロックが8Kバイト又は16Kバイトとされるが、1ブロック=8Kバイトとされる板状メモリ15の場合は、FATで扱うクラスタは8Kバイトとされる。また1ブロック=16Kバイトとされる板状メモリ1の場合は、FATで扱うクラスタは16Kバイトとされる。即ち、8Kバイト又は16KバイトがFAT管理上のデータ単位であり、かつ板状メモリでのブロックとしての1つのデータ単位とされる。従って板状メモリからみれば、FATで扱われるクラスタサイズ=その板状メモリのブロックサイズとなる。このため、本例の以降の説明については、簡略化のためにブロック=クラスタとして考えることとする。

【0129】そしてブロックナンバとしてx・・・(x+m-1)、(x+m)、(x+m+1)、(x+m+2)・・・と示したが、例えばこのように各ブロックにおいてFAT構造を構築する各種データは記憶されることになる。なお、実際には必ずしもこのように物理的に連続する各ブロックに各情報が記憶されるものではない。

【0130】FAT構造において、まずパーティションテーブルには、最大2GバイトとされるFATパーティションの先頭と終端のアドレスが記述されている。ブート領域には、いわゆる12bitFAT、16bitFATの別や、FAT構造として大きさ、クラスタサイズ、各領域のサイズなどが記述される。

【0131】FATは、後述するように各ファイルを構成するクラスタのリンク構造を示すテーブルとなり、またFATについては続く領域にコピーが記述される。ルートディレクトリには、ファイル名、先頭クラスタ番

号、各種属性が記述される。これらの記述は1つのファイルにつき32バイト使用される。

【0132】[板状メモリのファイル構造]

【ディレクトリ構成】次に、板状メモリに記憶されるファイル構造について説明していく。まずディレクトリ構成例を示す。上述したように、板状メモリで扱うことのできる主データとしては、動画データ、静止画データ、スピーチ等のオーディオデータ、音楽用データ等のCDやMDなどから再生される高音質なオーディオデータ、制御用データなどがあるが、このためディレクトリ構造としては、ルートディレクトリから、VOICE（スピーチ用ディレクトリ）、DCIM（静止画用ディレクトリ）、MOxxxxnn（動画用ディレクトリ）、AVCTL（制御用ディレクトリ）、HIFI（音楽用ディレクトリ）が配される。

【0133】本例では、特にオーディオデータである（スピーチデータ）のファイルについて詳しく述べていくこととする。ディレクトリVOICEのサブディレクトリとしては、オーダーファイル（ORDER.MSF）、付加情報管理ファイル（INFO.MSF）、フォルダ（FOLDER1、FOLDER2・・・）等が形成される。また、例えばフォルダ内には、実際のオーディオデータであるスピーチデータのファイル（ファイル名98120100.MSV等）が形成される。

【0134】[スピーチデータファイル] スピーチデータファイルのデータ構造を示す。スピーチデータファイルは、フォーマットフレーム（FORMAT FRAME）、TOCフレーム（TOC FRAME）、タイトルフレーム（TITLEFRAME）、メーカーフレーム（MAKER FRAME）、オーサーフレーム（AUTHOR FRAME）、スペースフレーム（SPACE FRAME）、データフレーム（DATA FRAME）を含んで構成される。但し、タイトルフレーム（TITLE FRAME）、メーカーフレーム（MAKERFRAME）、オーサーフレーム（AUTHOR FRAME）を設けることは任意であり、基本的には、太枠で囲ったフォーマットフレーム（FORMAT FRAME）、TOCフレーム（TOC FRAME）、スペースフレーム（SPACE FRAME）、データフレーム（DATA FRAME）によってスピーチデータファイルが構成されることになる。

【0135】フォーマットフレームは、当該スピーチデータファイルの基本的な管理情報となり、コーデックの種類等が示される。TOCフレームは、当該スピーチデータファイルの各フレームの配列状態を表す管理情報となる。つまりTOCフレームの記述によりスピーチデータファイルのフレーム構造が識別される。データフレームは、実際のスピーチデータが格納される領域となる。スペースフレームは、詳しくは後述するが、未使用のエリアとされ、これがTOCフレームの拡大の場合の予備

領域や、ファイル内の再生不可エリアの設定のために機能する。

【0136】このような各フレームにおいて、データフレームとスペースフレームは同一ファイル内に複数個存在する場合もある。ただし、データフレームとスペースフレーム以外のフレームは、同一ファイル内に1つとなる。また、フォーマットフレームは必ずファイルの先頭に配置される。そしてTOCフレームはフォーマットフレームの直後に配置される。上述した必須でないフレームであるタイトルフレーム、メーカーフレーム、オーサーフレームは、TOCフレームの後にまとめて配置される。これらはフレームIDが小さい順番に配置されることになる。また、必須でないフレームの並びの次には、必ずスペースフレームが配置される。

【0137】また、スピーチデータファイルのデバインド処理時には、分割ポイントが含まれるクラスタにおいて、再生されないデータエリアをスペースフレームとする。即ち、或るクラスタの途中の位置が分割点とされた場合、分割による後半のファイルでは、スピーチデータの先頭が、そのクラスタの分割点となる。あくまでファイルの最小単位はクラスタとなるため、このようにそのクラスタ内の分割点以前のデータは、再生してはいけないデータが含まれることとなるが、その部分は、スペースフレームとして再生されないようにする。

【0138】また、スピーチデータファイルの分割、結合処理の結果、2つのスペースフレームがとなり合う状況になる場合もあり得る。このようにとなり合う2つのスペースフレームが存在する場合は、それを1つのスペースフレームにまとめる。つまり具体的には、1つのスペースフレームとしてTOCフレームで管理されるようにする。

【0139】以上、実施の形態としての携帯情報端末および板状メモリとこれらによるフラッシュROM書き換え動作について述べてきたが、本発明はこれらの構成及び動作に限定されるものではない。特に実施の形態においては板状メモリを用いたシステムにおいて主データがオーディオデータとしてのボイスデータであるオーディオデータファイルまたはボイスデータファイルの書き込み、読み出し、消去のためのROMデータの書き換え処理を例にあげたが、上述したように板状メモリを用いたシステムではボイスデータファイル以外に、音楽データファイル等のオーディオデータファイル、動画データファイル等の画像データファイルなども扱うことができる。そしてこのオーディオデータファイル、画像データファイルについても、上記同様にROMデータの書き換え処理が実行されることで、携帯情報端末20のマイクロコンピュータの動作を安定化できる。

【0140】このようにして、ファイルシステムとしてFATファイルシステムを使用して、記録媒体として脱着機構を介してフラッシュメモリを使用し、スピーチ・

音楽等のオーディオデータであるストリームデータを記録または再生する機能を有し、ROMデータの書き換えが可能なマイクロコンピュータを有する携帯情報端末において、何等マイクロコンピュータを分解することなく、ROMデータを書き換えることが可能となり、携帯情報端末20のマイクロコンピュータの動作の安定化を図ることができ、ROMデータの更新時や修理時における操作性および保守性の向上を図ることができる。

【0141】なお、上述において、FAT上あるいはFATファイルシステム上とあるのは、すべて板状メモリを構成する不揮発性メモリ内におけるファイル管理システムを意味している。また、本実施の形態の携帯情報端末に適用される記録媒体としては、図2のような板状メモリに限定されるものではなく、他の外形形状とされた固体メモリ媒体としてメモリチップ、メモリカード、メモリモジュール等や例えば、Siemens、Sandisk、日立、Motorola、NEC、Nokiaの提唱するMultiMediaCard (MMC) や、東芝、松下、Sandiskが提唱するSDメモリカード (SD [Secure Digital] Memory Card)、あるいはPCカード等でも構わない。

【0142】また、これまで説明したファイルシステムのフォーマットも、FATファイルシステムに限定されるものではなく、他のファイルシステムでも良く、また、例えば実際に応じてその細部の規定などは変更されて構わない。更には、フラッシュメモリ容量のバリエーションも上述したものに限定されるものではない。もちろん、本実施の形態の記憶媒体のメモリ素子はフラッシュメモリに限られず、他のメモリ素子や、書き替え可能なコンパクトディスクでもよい。

【0143】また、本発明における不揮発性メモリであるフラッシュROM2の書き換えを可能にするための機能のすべてを板状メモリ7から供給するのではなく、フラッシュROM2の書き換えを可能にするための機能の一部のみを板状メモリ7から供給するようにしても良い。

【0144】

【発明の効果】本発明の携帯情報端末装置は、本体に対して脱着可能に設けられ、プログラムデータの書き換えを実行する書き換えプログラムを記憶した記録媒体と、制御手段に対して、記録媒体の装着時に書き換えプログラムに基づいて書き換え禁止手段の書き換え禁止状態を少なくとも1回解除するように制御する解除手段とを備えたので、プログラムデータの書き換えを可能にするための機能は、記録媒体に記憶した書き換えプログラムの中に設けることにより、この書き換えプログラムが制御手段の中に読み込まれて、初めてプログラムデータの書き換え禁止状態を解除して書き換え可能状態に制御され、この書き換えプログラムにより、新たなプログラム

データを制御手段により本体内部に書き込むことにより、プログラムデータの書き換えができるため、何等制御手段としてのマイクロコンピュータを分解することなく、プログラムデータを書き換えることが可能となり、携帯情報端末のマイクロコンピュータの動作の安定化を図ることができ、プログラムデータの更新時や修理時における操作性および保守性の向上を図ることができるという効果を奏する。

【図面の簡単な説明】

10 【図1】実施の形態のファイルシステム処理階層の説明図である。

【図2】実施の形態の携帯情報端末および板状メモリの構成を示す図である。

【図3】実施の形態のマイクロコンピュータおよび板状メモリの間でのデータの流れを示す図である。

【図4】実施の形態のフラッシュROM書き換え動作を示すフローチャートである。

20 【図5】実施の形態の必要なデータの作成動作を示すフローチャートである。

【図6】実施の形態の物理アドレスファイルの構成を示す図である。

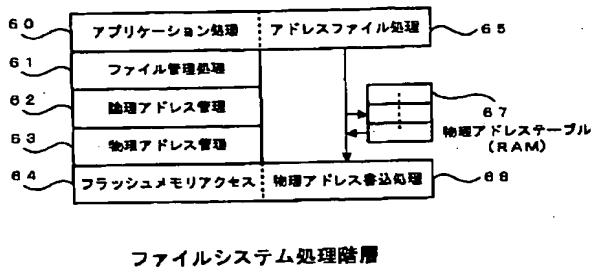
【図7】実施の形態のヘッダーデータの例を示す図であり、図7Aは識別用ファイルのヘッダーデータでありパーソナルコンピュータからのロードの場合、図7Bは識別用ファイルのヘッダーデータであり板状メモリからのロードの場合、図7Cおよび図7DはプログラムローダーファイルおよびROMデータファイルのヘッダーデータである。

30 【図8】実施の形態のフラッシュROMの書き換えを行うプログラムが記憶された板状メモリ内部のデータの記憶状態を示す図である。

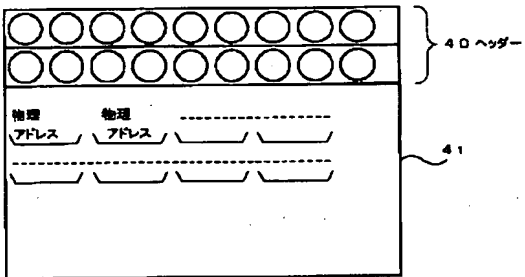
【符号の説明】

1……マイクロコンピュータ、2……フラッシュROM、3……内部RAM、4……ライトイネーブル端子、5……出力端子、6……外部RAM、7……板状メモリ、8……ローダー用板状メモリ識別用ファイル、9……プログラムローダーファイル、10……プログラムローダーアドレスファイル、11……ROMアドレスファイル、12……ROMデータアドレスファイル、13……脱着機構、14……ジャンパー線、20……携帯情報端末 (セット本体)、40……ヘッダー、41……物理アドレス、50……識別データ、51……スペース、52……1ブロックあたりのページ数、53……識別データ、54……スペース、55……1ブロックあたりのページ数、56……識別データ、57……ファイル名、58……スペース、59……物理ブロックアドレスデータ、60……アプリケーション処理層、65……アドレスファイル処理部、67……物理アドレステーブル (RAM)、56……物理アドレス書き込み処理部、

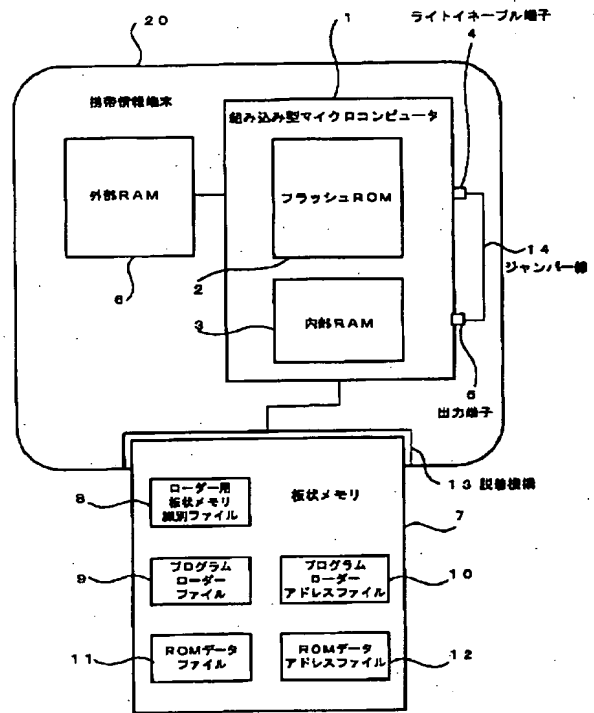
【図1】



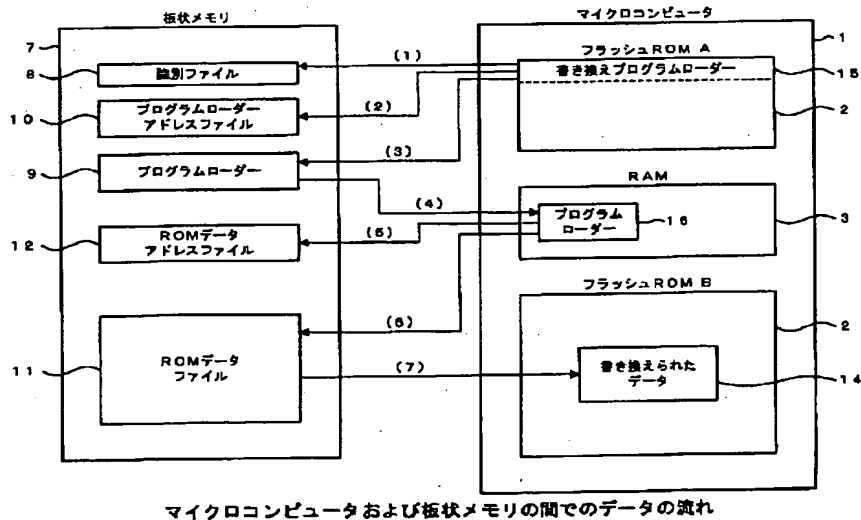
【図6】



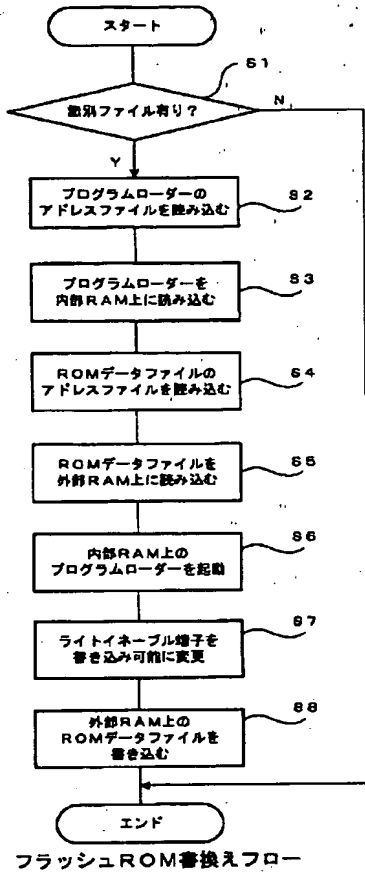
【図2】



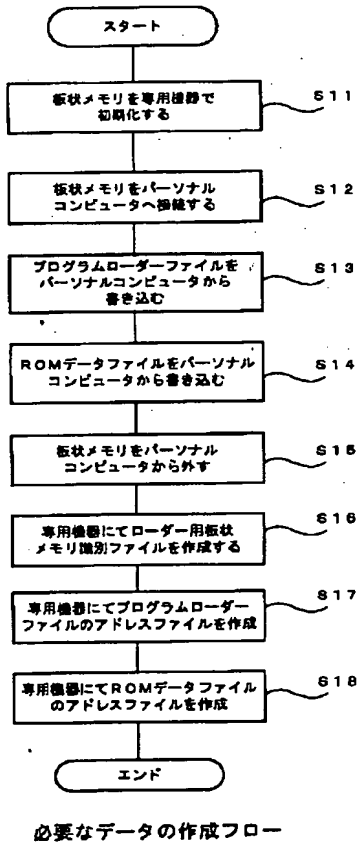
【図3】



【図4】



【図5】



【図7】

パーソナルコンピュータからのロードの場合

	0x00	0x01	0x02	0x03	0x04	0x05	0x06	0x07	0x08	0x09	0x0A	0x0B	0x0C	0x0D	0x0E	0x0F
A	0x0000	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
	0x0010	P	Q	R	S	T	U	V	W	X	Y	Z	AA	AB	AC	AD
	0x0020	L	O	A	D	F	R	O	M	P	C					

板状メモリからのロードの場合

	0x00	0x01	0x02	0x03	0x04	0x05	0x06	0x07	0x08	0x09	0x0A	0x0B	0x0C	0x0D	0x0E	0x0F
B	0x0000	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
	0x0010	P	Q	R	S	T	U	V	W	X	Y	Z	AA	AB	AC	AD
	0x0020	L	O	A	D	F	R	O	M	P	C					

識別用ファイルのヘッダーデータ

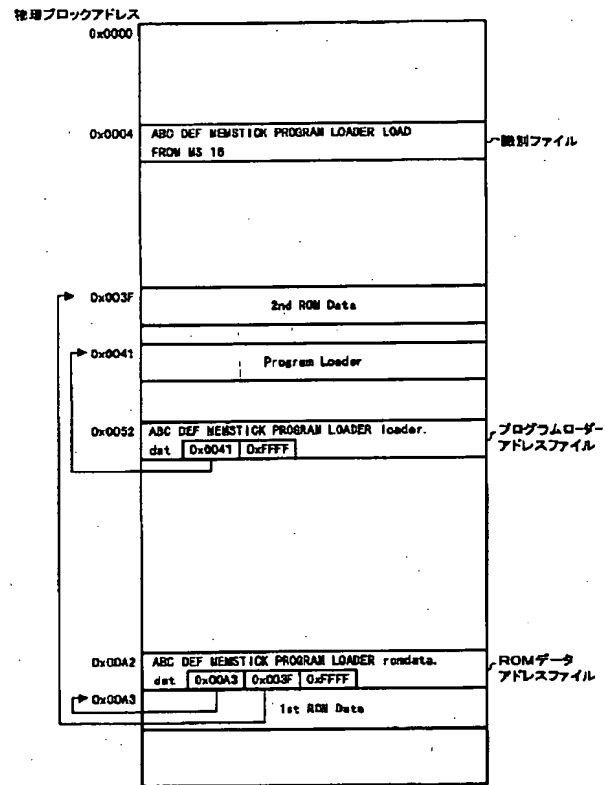
	0x00	0x01	0x02	0x03	0x04	0x05	0x06	0x07	0x08	0x09	0x0A	0x0B	0x0C	0x0D	0x0E	0x0F
C	0x0000	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
	0x0010	P	Q	R	S	T	U	V	W	X	Y	Z	AA	AB	AC	AD
	0x0020	r	o	m	d	a	t	a								
	0x0030	0x00A3	0x003F	0xFFFF												

プログラムブロックアドレスデータ

	0x00	0x01	0x02	0x03	0x04	0x05	0x06	0x07	0x08	0x09	0x0A	0x0B	0x0C	0x0D	0x0E	0x0F
D	0x0000	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
	0x0010	P	Q	R	S	T	U	V	W	X	Y	Z	AA	AB	AC	AD
	0x0020	I	O	A	D	F	R	O	M	P	C					
	0x0030	0x0041	0xFFFF													

プログラムローダーファイルおよびROMデータファイルのヘッダーデータ

【図8】



フラッシュROMの書き換えを行うプログラムが
記憶された板状メモリ内部のデータの記憶状態

フロントページの続き

(51) Int. Cl.⁷

G 0 6 K 19/07

識別記号

F I

G 0 6 K 19/00

ターマコード(参考)

N